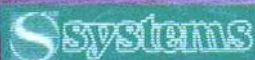
Anno VIII - N. 63 - 25 aprile 1989
Sped. Abb. Post. Gr. III/70 CR - Distr.: MePe

Amiga, la tua prima fotocamera digitale

C64 IN CONCERTO

- Sette note in LM
- Tutte le scale delle tue canzoni

AMIGA

Mixer d'immagini per videomani

C 128

Orologio permanente

Pool of radiance, il re dei giochi per 64

Sommario

RUBRICHE

Direttore: Alessandro de Simone - **Caporedattore:** Michele Maggi
Redazione/collaboratori: Paolo Agostini, Davide Ardizzone, Claudio Baiocchi, Angelo Bianchi, Luigi Callegari, Sergio Camici, Umberto Colapicchioni, Maurizio Dell'Abate, Valerio Ferri, Roberto Ferro, Cristina Magnaghi, Giancarlo Mariani, Roberto Marigo, Clizio Merli, Marco Mietta, Marco Miotti, Oscar Moccia, Roberto Morassi, Guido Pagani, Antonio Pastorelli, Sonja Scharrer, Fabio Sorgato, Valentino Spataro, Danilo Toma, Franco Rodella, Stefano Simonelli
Grafica: Arturo Ciaglia
Direzione, pubblicità: via Mosè, 18 - 20090 Opera (MI) - Tel. 02/55500310 - **Redazione:** Tel. 02/5249211
Pubblicità: Milano: Leandro Nencioni (direttore vendite), Angelo Ricupero - Via Mosè, 18 - 20090 Opera (MI) - Tel. 02/55500310
• Emilia Romagna: Spazio E - P.zza Roosevelt, 4 - 40123 Bologna - Tel. 051/236979
• Toscana, Marche, Umbria: Mercurio srl - via Rodari, 9 - San Giovanni Valdarno (Ar) - Tel. 055/947444
• Lazio, Campania: Spazio Nuovo - via P. Foscari, 70 - 00139 Roma - Tel. 06/8109679
Segreteria: Tiziana Sodano - **Abbonamenti:** Liliana Spina (Tel. 02/55500355)
Arretrati e software: Lucia Dominioni (Tel. 02/8467348/9 - viale Famagosta 75 - 20142 Milano)
Tariffe: prezzo per copia L. 5.000. Abbonamento annuo (11 fascicoli) L. 50.000. Estero: il doppio.
Abbonamento cumulativo alle riviste Computer e Commodore Computer Club L. 90.000.
I versamenti vanno indirizzati a: Systems Editoriale Srl mediante assegno bancario o utilizzando il c/c postale n. 37952207
Composizione: Systems Editoriale Srl - **Fotolito:** Systems Editoriale Srl
Stampa: Systems Editoriale/La Litografica Srl - Busto Arsizio (Va)
Registrazioni: Tribunale di Milano n. 370 del 2/10/82 - Direttore Responsabile: Michele Di Pisa
Sped. in abb. post. gr. III - Pubblicità inferiore al 70% - **Distrib.:** MePe - via G. Carcano, 32 - Milano
Periodici Systems: Banca Oggi - Commodore Club (disco) - Commodore Computer Club - Commodore Computer Club (disco produzione tedesca) - Computer - Computer disco - Electronic Mass Media Age -Energy Manager - Hospital Management - Jonathan - MondoRicambi - Nursing '90 - PC Programm (disco) - Personal Computer - Security - Software Club (cassetta ed. italiana) - TuttoGatto - Videoteca - VR Videoregistrare

L'UTENTE DELLA SECONDA GENERAZIONE

In laboratorio vengono sperimentati processori straordinari che vedremo sul mercato solo fra alcuni anni; oggi, invece...

E' noto che il progresso tecnologico nel campo dell'hardware è di gran lunga più avanti di quello del software, sia questo di base (sistemi operativi, linguaggi) che di applicazioni generali (pacchetti applicativi).

Volendo restare nel campo Ms-Dos, ad esempio, mentre le industrie si affannano a lavorare sul processore quasi sperimentale 80486, la pubblicità spinge all'acquisto di macchine basate sull'80386, la IMB (sembra) un po' più sicura di prima nel dare fiducia al sistema PS/2 (basato sull'80286), il mercato vede con piacere cifre soddisfacenti per ciò che riguarda programmi che girano su 8088/86, e l'utente smanettone "medio" (buon ultimo!) incomincia finalmente a capirci qualcosa di questo vecchio, ma attuale, 8088/86.

Dunque (ed anche i più ottimisti non potranno che darci ragione) l'hardware attualmente prodotto (cioè l'80386) si trova "avanti" di ben due lunghezze (80286 - 8088) rispetto all'attuale fascia di utilizzo "medio" dei Personal Computer professionali. Figuriamoci in che posizione si trovano i processori sperimentali.

Tale particolare condizione non si verifica per disinformazione o mancanza di sensibilità da parte di software house o utenti; nè è da attribuire allo scarso impegno delle une e degli altri.

Il "gap" è dovuto esclusivamente all'inevitabile periodo di tempo indispensabile, all'uomo, per comprendere appieno le potenzialità di un micro in modo da sfruttarle adeguatamente.

Una stessa motocicletta, seguendo una banale analogia, offre diverse prestazioni a seconda di chi la governa. Probabilmente alcune particolari prodezze non erano previste dai progettisti nè sono comunemente intuibili finchè non assistiamo ad una gara motociclistica o a determinate esibizioni di impavidi Stuntmen.

"Fatto il computer, trovato l'impiego", si può dire parafrasando il celebre detto "Fatta la legge, trovato l'inganno".

L'inganno, semmai, è il voler insistere oltre misura nello studio approfondito fine a se stesso, nella ricerca maniacale del millisecondo in più, rilevabile solo da strumentazione adeguata, non già dall'uomo, cui è destinata la macchina.

Quando il C/64 fu costruito, nessuno lo conosceva a fondo, forse nemmeno gli stessi progettisti. Ecco, quindi, che apparvero le prime pubblicazioni relative all'aggiunta di comandi Basic, alla manipolazione dei vettori, alle tecniche di alterazione del Raster, dell'interrupt.

E poi, dopo la "scoperta" del drive, giunsero puntuali le richieste di esplorazione di tracce e settori, di allocazioni inconsuete, di autorun originali, di protezioni esclusive.

Da un po' di tempo giungono invece, altrettanto puntualmente, richieste di programmi utili, pratici, semplici, che possano, in altre parole, individuare nel computer lo strumento che dovrebbe essere: un aiuto concreto nelle più disparate situazioni che troviamo abitualmente a scuola (matematica, archivi, problemi di vario tipo, ricerche), nel lavoro (data base, word processor, spreadsheet, gestioni varie, comunicazione), a casa nostra (divertimento, gestione domestica, condominio, spese e risparmi).

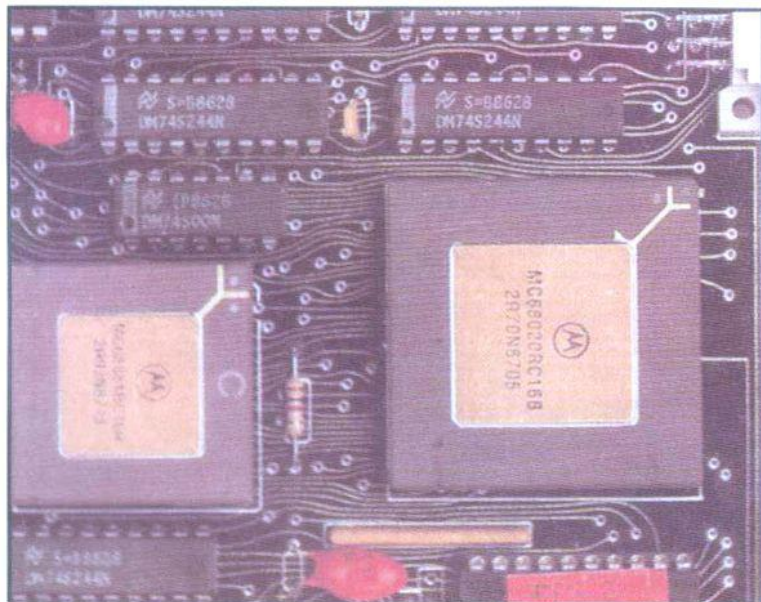
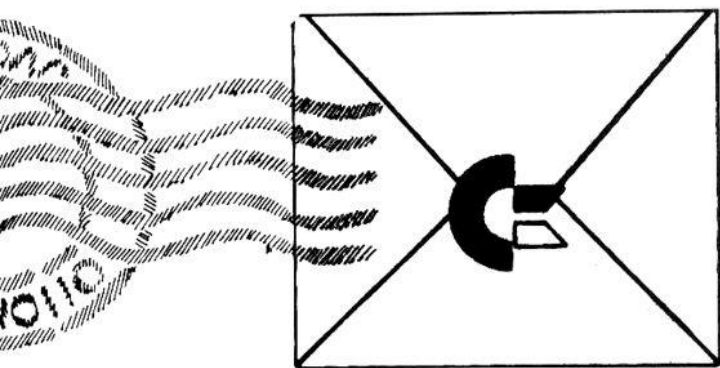
Ecco, dunque, che dopo aver conosciuto la macchina si "pretende" di utilizzarla.

Ed è proprio questo il motivo per cui dovrebbero esser privilegiati argomenti, programmi e subroutine che, più che esplorare l'interno del computer, tendano ad utilizzarlo per quello che è, e deve esser considerato: un elettrodomestico al nostro servizio.

Abbiamo oggi, insomma, raggiunto il livello di conoscenze adeguato alle potenzialità offerte dal micro 6510, e dal C/64 in particolare. E' giusto, quindi, che tali conoscenze vengano messe a frutto; in Basic o linguaggio macchina, in overlay o in Pascal; per giocare o studiare, magari per lavorare(!); e, in ogni caso, per rispondere adeguatamente alle esigenze dell'utente della "seconda" generazione.

Contemporaneamente dovrebbe aver inizio l'altra fatica, relativa alla conoscenza del 68000 e della "macchina" Amiga; proprio come, qualche anno fa, iniziammo con l'inesauribile C/64.

E' proprio vero: nel campo dell'informatica non si finisce mai...

Alessandro de Simone

la vostra posta

SENZA SYSTEM

☐ E' possibile eliminare le "System Request" dell'Amiga, soprattutto lavorando in AmigaBasic?
(Stefano Troncia - Carloforte)

• Le "Richieste di sistema", citate dal nostro lettore, sono indispensabili per il corretto funzionamento di molti programmi e non solo nel caso di AmigaBasic.

A volte, infatti, l'elaborazione non può proseguire se non vengono attivati determinati programmi presenti su altri dischetti.

Un tipico esempio è rappresentato dal comando Basic "Say" (in grado di sintetizzare la voce umana) che, se presente in un programma, impone al computer di rintracciare il file idoneo. A questo punto, infatti, compare la richiesta di inserimento del disco di Workbench che, di solito, contiene il file richiesto.

E' intuitivo che, in casi come questi (in cui, cioè, viene esplicitamente richiesto il caricamento di un file) il lavoro non può procedere se non si soddisfa la "Richiesta di sistema".

Se si dispone di due drive, in uno dei quali è sempre presente il dischetto contenente i file necessari (Workbench, nel caso appena visto), la "System Request" non viene nemmeno visualizzata dal momento che Amiga si "accorge" della presenza del floppy e provvede automaticamente a gestire l'intera procedura.

DOV'E' L'ERRORE?

☐ Ho scritto un banale programma in Basic, ma probabilmente è presente un errore "logico" che, in quanto tale, non riesco ad individuare. Potete aiutarmi a cercarlo?
(Filippo Piscopo - Messina)

• Il programma inviato dal nostro lettore è un semplice archivio che agisce sul drive.

Dopo aver inserito i vari nominativi, un opportuno comando...

```
Open #2, 8, 2, "Dati, s, w"
```

...provvede ad aprire, in scrittura su disco, il file "Dati". Il primo dato ad esser trascritto è, giustamente, il numero di dati (variabile A) che verranno tra breve registrati...

```
Print #2, A
```

...In seguito, mediante un ciclo For... Next, vengono effettivamente trascritti i singoli dati:

```
For R=1 to A
Print #2, Nome $(R)
Next
```

Al momento della successiva lettura dovrebbe esser caricato il valore A mediante l'idonea istruzione che è...

```
Input #2, A
```

...e non...

```
Get #2, A
```

...perchè, in quest'ultimo caso, viene letto il solo... segno del numero "A" precedentemente memorizzato.

Quando, infatti, si apre un file in modo sequenziale, un qualsiasi valore numerico viene memorizzato come una sequenza, appunto, di caratteri e non secondo un codice numerico.

Ad esempio, il numero -879.56 viene registrato come sequenza di sette caratteri, ed esattamente "-", "8", "7", "9", ".", "5", "6". Tentando di leggere questo numero mediante GET#, e non mediante INPUT#, si "cattura", in effetti, solo il segno.

Una trattazione approfondita sulla gestione dei file sequenziali su nastro e su disco (che ha occupato l'intero inserto di 16 pagine) è stata affrontata sul N. 39 di Commodore Computer Club.

C/64 E INTERFACCIA MIDI

☐ Sono in possesso di una tastiera musicale dotata di interfaccia MIDI e vorrei collegarla al mio C/64. Esiste in commercio software ed hardware adatti allo scopo?
(Anonimo)

• Abbiamo più volte affrontato il discorso MIDI e possiamo assicurare il nostro lettore che sono in commercio, per il C/64, interfacce e programmi più che validi.

Ti consigliamo di rivolgerti ad un ottimo negozio specializzato in strumentazione musicale, oppure di contattare l'importatore della tua tastiera che, ne sono sicuro, potranno aiutarti nella ricerca.

VOCABOLARIO PER C/64

☐ Sto scrivendo un programma ("Vocabolario") che dovrebbe aiutare gli studenti nella traduzione inglese dei vocaboli italiani (e viceversa). Dal momento che è probabile che la sua occupazione richiederà oltre 220 blocchi, come potrò caricarlo sul mio C/64?

• Ognuno dei 664 blocchi di un disco è in grado di memorizzare 256 byte. 220 blocchi, pertanto, rappresentano oltre 56000 byte che non possono essere "ospitati" dal C/64. Sebbene questo, infatti, disponga di 64 KRam, una parte della memoria "deve" esser destinata al programma in grado di gestire i vari dati.

I programmi più lunghi devono quindi necessariamente essere caricati in più blocchi, da far funzionare uno alla volta.

Di conseguenza, un archivio (quale è un vocabolario) non potrà mai essere ospitato per intero nella memoria di un C/64.

Supponiamo, infatti, che ogni vocabolo occupi, in media, 10 byte (e altrettanti ne occupi la corrispondente traduzione); lavorando in Basic, e supponendo di avere a disposizione 30 Kram (il resto sarà destinato al programma), potrai memorizzare 30000 / 20 = 1500 vocaboli, che non sono poi tanti.

Tenendo inoltre presente che un vocabolario, per essere efficiente, deve presentare vari sinonimi ed esempi d'uso, il numero dei vocaboli disponibili diminuisce paurosamente.

Ecco il motivo per cui un vocabolario, appena "decente", può esser gestito solo su disco dal momento che questo può ospitare quasi 170 mila caratteri. La lentezza del 1541, però, sconsiglia il sistema C/64 - 1541 per un lavoro così impegna-

AIUTATECI A SERVIRVI MEGLIO

Spesso alcuni lettori porgono quesiti le cui risposte sono già state esplicitamente pubblicate (in occasione di risposte ad analoghe domande) oppure sono contenute in articoli presenti nei fascicoli in loro possesso.

Per evitare di ripetere argomenti già trattati, pertanto, ricordate di indicare sempre, nelle lettere che inviate, i numeri dei fascicoli che possedete: potremmo infatti indicarvi, se esistono, gli articoli che, in un modo o nell'altro, chiariscono gli argomenti richiesti.

Si ricorda ai lettori che non ci è possibile rispondere privatamente, nemmeno se si acclude l'affrancatura per la risposta.

Per accelerare il servizio, ricordate di indicare sempre la data di spedizione dal momento che questa costituisce diritto di precedenza.

A tutt'oggi (con la pubblicazione del presente fascicolo) risulta evasa l'intera corrispondenza pervenuta in Redazione entro la fine del mese appena trascorso.

Coloro che non vedessero citato il loro nome nella presente rubrica, possono trovare risposta ai loro quesiti in appositi articoli, apparsi nel frattempo, o di prossima pubblicazione.

La mancanza di adeguata risposta può esser dovuta anche al fatto che gli argomenti richiesti sono sviluppati in modo esauriente nei libretti di istruzione allegati ai computer, alle periferiche o al software originale.

Non ci è possibile dare consigli su acquisti da effettuare nè esprimere pareri estranei alla logica della correttezza professionale e commerciale.

tivo: affinchè il programma sia realmente efficiente è indispensabile una ricerca rapidissima, che può essere svolta solo da un hard disk.

La morale della favola è praticamente scontata; per applicazioni professionali (ed un vocabolario è tra queste), è opportuno cambiar computer.

I POLLICI DI MS-DOS

□ Ho un suggerimento da darvi: divulgare il software per Ms-Dos (che pubblicate su "Commodore Computer Club" e su "Personal Computer") sia su dischetti da 5.25 sia su quelli da 3.5 pollici.
(Benedetto Chinaglia - Bolzano)
(Luca Boccaccini - Ravenna)

• Dateci il tempo di organizzarci adeguatamente e vedrete che qualcosa di concreto proporremo nei prossimi mesi.

Nel frattempo (spudorata pubblicità) acquistate "Personal Computer" che è la testata cui far riferimento per i sistemi Ms-Dos.

GO TO

□ Ho notato, esplorando le Rom del mio C/64, che oltre ai comandi Goto (codice 137) e Gosub è presente anche un misterioso e semplice Go (codice 203). Forse si tratta di un comando sconosciuto (del tipo GO 64 del C/128) presente in Rom, ma non implementato?
(Carmelo Cipolla - Caltanissetta)

• La giusta osservazione del lettore viene del tutto chiarita grazie ai due microlistati che seguono, da digitare così come sono, e che subito passiamo a commentare.

Il primo di questi...

```
100 go to 110
110 rem aaaaaaa
120 for i=2053 to 2059
130 print peek(i);: next
```

...provvede a leggere se stesso dal momento che, a partire dalla locazione 2053, è presente il codice della prima istruzione della prima riga (cioè di riga 100).

Facendo partire il programma con il solito Run (e supponendo che sia presente uno spazio dopo GO e dopo TO) sullo schermo compaiono sette valori...

```
203, 32, 164, 32, 49, 49, 48
```

...che altro non sono se non il codice di GO (203), quello di spazio (32), quello di TO (164) e poi ancora lo spazio, il codice del carattere "1" (cioè 49), ancora "1" ed infine "0" (48).

Se, invece, viene modificata la riga 100 come segue...

```
100 goto 110
```

...vale a dire scrivendo GOTO senza spazio tra GO e TO, la sequenza di valori viene modificata come segue:

```
137, 32, 49, 49, 48, 0, 25
```

Ciò significa che al comando GOTO è ora associato un unico valore (137) e non due, come nell'esempio di prima.

In conclusione il codice GO è presente perchè la sintassi del Basic consente entrambe le forme ("Go To" e "Goto") e l'interprete deve pur essere in grado di distinguere tra le due prima di... emettere un Syntax Error!

SCOPERTE

□ Ritengo di aver scoperto una procedura interessante che riguarda gli utilizzatori del vostro "Emulatore di Gw-Basic".
(Lettore anonimo)

• "Ho notato -prosegue il distratto lettore- che tutte le volte che si incappa in una condizione di errore il programma si arresta e viene automaticamente attivato lo

LA VOCE DEI LETTORI

OPINIONI

La rubrica "La voce dei lettori" è nata per dare spazio (e sfogo...) ai lettori che, pur non avendo nulla di particolare da chiedere, intendono far sentire la propria voce, chiedere solidarietà ad altri utenti o, semplicemente, riportare osservazioni ritenute utili o importanti. Per questioni di spazio le lettere sono spesso condensate; vengono poi "censurate" le osservazioni ed i termini troppo "forti" che possano arrecare offesa ai lettori, agli inserzionisti e... a noi stessi. In calce viene riportata la sola iniziale del cognome del lettore (salvo casi particolari), per fare in modo che questi possa sentirsi ancor più libero di esprimere la propria opinione. Coloro che desiderano il minor numero di "tagli" possibile alle loro missive sono pregati di inviarle su DISCO, usando il word processor Easy Script.

Screen 0. Ciò è utile nel caso in cui si lavori in bassa risoluzione ma non può essere accettato lavorando in alta risoluzione.

Disassemblando "Gw-Basic" ho notato che è sufficiente impedire l'attivazione di Screen 0 mediante una semplice...

Poke 40576, 96

...che consiste nel sostituire il codice NOP, ivi presente, con un RTS. In questo caso, nonostante vi sia un errore, si "rimane" in hi res ed il comando On Error Goto funziona senza esser costretti a ritornare sempre in low-res.

Per ripristinare le condizioni iniziali è sufficiente...

Poke 40576, 234

In questo modo, però, il comando Screen 0 non funziona più correttamente. Per ovviare all'inconveniente è sufficiente digitare...

Sys 40585

...per tornare in Low-res".

IN ATTESA DI GIUDIZIO

□ Vi invio due miei programmi per sottoporli ad un vostro giudizio.
(Andrea Faro - Catania)

• Dalla cura che hai dedicato alla stesura dei listati si nota una certa precisione, soprattutto per ciò che riguarda l'aspetto estetico delle varie schermate di spiegazione.

Ti consiglio, però, di fare in modo che la prosecuzione dipenda dalla pressione di un tasto da parte dell'operatore e non da un ciclo For... Next "a vuoto".

Se, infatti, all'inizio è necessario un certo intervallo di tempo per far comprendere il messaggio apparso sul video, in seguito l'attesa non è più necessaria.

Ad esempio, invece di...

```
2000 For X=1 to 32000: Next
```

...ti consiglierei:

```
2000 For X=1 to 32000
2010 Get X$: If X$ = Chr$ (13) then X
= 32000
2020 Next
```

In questo modo, pertanto, una schermata rimane sul video per il tempo stabilito a meno che non si prema il tasto Return (codice 13). In questo caso, infatti, il ciclo For Next viene interrotto dal momento che X raggiunge forzatamente il limite massimo imposto.

Per ciò che riguarda l'altro programma ("Archivio Elicotteri"), che senza dubbio è originale, tieni conto che si dovrebbe ricorrere al computer solo nei casi in cui, con carta e penna, non sia possibile (o rapido) raggiungere gli stessi risultati.

Il programma inviato non fa altro che visualizzare i dati tecnici (e varie altre curiosità) relativi a numerosi elicotteri. Lo stesso risultato si può ottenere più facilmente ricorrendo ad un album illustrato.

La strada intrapresa, comunque, è certamente quella giusta: continua così e vedrai che, prima o poi, raggiungerai risultati degni di nota.

LA VOCE DEI LETTORI

GIU' LE MANI DAL C/128

No, per favore, non toccate il mio caro C/128, soprattutto per quanto mi è costato (2 milioni 400 mila, completo di tutto, compreso un TV color Sony), pur non avendolo acquistato a rate.

E poi qualsiasi critica fatta a tale macchina non può essere altro che crudele e calunniatrice. A mio parere l'apparecchio è valido esteticamente, ha un'ottima capacità di memoria, può sfruttare tutte le periferiche ed i programmi del C/64, funziona in tre modalità etc. etc. etc.

Dunque, cos'altro pretendere da questo "poveraccio" se non è stato predisposto per servirci anche l'espresso?

E' vero, tutti lo sanno (nè intendo portare vasi a Samo) che in questo campo bastano due mesi per rendere obsoleto anche il più potente elaboratore; ma ciò non equivale a dire che quanto acquistato due mesi prima sia da buttar via solo due mesi dopo. Io sono d'accordo con l' "Ubi maior, minor cessat" dei latini... ma non a tal punto. Voglio dire, in altre parole, che la Prisma del signor de Simone rimane tale anche quando viene sorpassata o parcheggiata da o vicino ad una Thema. Nè voglio credere che il signor de Simone cambierebbe la sua Prisma se succedesse quanto sopra paventato.

Ecco che cosa ho capito leggendo il vostro giornale: il C/64 è nel cuore di tutti (per molti è stato il primo amore); ed il primo amore, si sa, non si può scordare. Il C/128, offrendo maggiori potenza e caratteristiche, riprende i motivi del C/64... e l'Amiga?

Standibus sic rebus non la compra nessuno? Ergo, disprezzando e sottovalutando i "piccoli" (che poi tanto piccoli non sono) si fa largo al Padrone, costringendo la clientela ad acquistare (magari disfacendosi del C/64-128) l'Amiga.

Tale comportamento, miei cari Amighi (pardon, Amici), a mio avviso è addirittura controproducente per la stessa Commodore perchè chi sente disprezzare un prodotto (magari da una rivista specializzata) si convince, inconsciamente, che realmente sia così; anzi, vedendolo disdegnare proprio da chi lo ha ideato (come fanno le belve con i nati) il potenziale acquirente viene assillato da un altro dubbio amletico: sarà buono (l'Amiga) oppure nasconde qualche difetto che, fra due mesi, mi sarà svelato?

Certo è che anche voi di C.C.C. parlando del C/128, mentre lo definite "potente computer", lo catalogate tra i "piccoli" di mamma Commodore e questo accade spesso...

Orbene, a me personalmente è sorto un altro dubbio di manzoniana memoria: "Fu vera gloria?"; se si fosse trattato di una "Prisma" l'avrei schedata in base alla cilindrata ma, in questo caso, come posso sapere se posseggo una 500 o una Ferrari Testarossa?

E' potente, sì o no, questo Basic 7.0 che correda l' "utilitaria"?
(Alfredo B.)

BANCA DATI E GESTIONE FAMILIARE

• Numerosi lettori lamentano malfunzionamenti di vario tipo nel corso delle elaborazioni dei programmi "Gestione Familiare" e "Banca Dati" da noi prodotti.

In TUTTI i casi segnalati gli errori si verificano a causa della superficiale lettura del libretto di istruzioni allegato al software.

Alcuni "particolari", inoltre, sfuggono inevitabilmente a coloro che si sono procurati il software piratato e non posseggono la necessaria documentazione per l'utilizzo ottimale dei due programmi citati.

ILLEGGIBILE

□ Alcune righe del listato "Digita uno vedi due" (vedi C.C.C. N. 60) risul-

LA VOCE DEI LETTORI

ROTTURE

Sono stato, fino a prima che mi capitasse ciò che sto per raccontare, un felice possessore di tutti i principali prodotti della fascia medio-bassa Commodore (Vic 20, C/64, Amiga 500).

I problemi che lamento riguardano l'assistenza in caso di guasto che, per un computer definito "professionale", dovrebbe essere quanto più professionale (appunto) e spedita possibile.

Il mio A-500, invece, si è rotto per un guasto all'integrato Fat-Agnus e da allora giace presso il centro di riparazione autorizzato della mia città. Sono seguite, ovviamente, numerose telefonate al centro stesso, che incolpava la Commodore di non fornire i pezzi di ricambio.

Mi chiedo se sia possibile che un computer, utilizzato prevalentemente per motivi di studio o lavoro, possa esser sottratto all'utilizzatore per periodi di tempo tanto lunghi.

Il computer professionale di mio padre (la cui marca non cito, ma è facilmente intuibile) fu riparato, in seguito ad un guasto, dalla sera alla mattina, grazie all'intervento di un tecnico giunto, addirittura, a domicilio!

(Domenico A.)

tano illeggibili. Potete ripubblicarle?
(Luca Vignale - Brandizzo)

• Eccole qua:

```
220 color 8, 15, 8: print: print: input "intervallo (x1-x2)"; x1, x2

230 if x2-x1 <= 0 then color 8, 4, 8: print: print: print tab(9) "errore: intervallo<=0!"
270 color 8, 14, 8: print: print: print tab (8) "sto calcolando min. e max"

390 locate 1, 2: print "x1 = "; locate 1, 5: print x1: locate 2, 2: print "x2 = "; locate 2, 5: print x2

400 locate 1, 22: print "max= "; locate 1, 26: print int ((l + .0005) * 1000) / 1000

410 locate 2, 22: print "min= "; locate 2, 26: print int ((m + .0005) * 1000) / 1000
```

Alcune righe risultano più lunghe di 80 caratteri. E' ovvio che, in questi casi, bisogna ricorrere alle abbreviazioni (punto di domanda invece di Print e così via).

DIFFERENZE

□ Vorrei conoscere la differenza esistente tra il sistema operativo CP/M e l'Ms-Dos e l'eventuale compatibilità tra i due.
(Fabrizio Santinelli - Camucia)

• L'obsoleto sistema operativo (S.O.) CP/M, "supportato" dal C/128, si basa sulla cosiddetta architettura dello Z-80 (microprocessore ad otto bit) e all'uso intensivo del drive, "visto" dal sistema come indispensabile area di memoria con la quale scambiar dati.

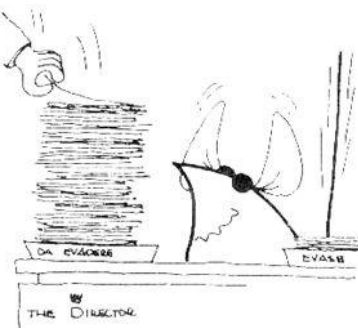
Il più moderno Ms-Dos si basa, invece, sul più attuale (ma obsoleto anch'esso) microprocessore 8088, diffuso anche in altre versioni, di cui la più nota è l'8086.

L'astro nascente, tuttavia, è il sistema operativo OS/2, basato sul più moderno, versatile e veloce microprocessore a 16 bit 80286.

Mentre, però, le precedenti versioni Ms-Dos girano correttamente anche sui moderni 80286 (ed i nuovissimi 80386), il CP/M risulta incompatibile con il S.O. Ms-Dos.

Ciò significa che programmi che girano "sotto" CP/M non possono funzionare sotto Ms-Dos.

ABBIAMO GIA' RISPOSTO

I seguenti lettori possono trovare adeguata risposta ai loro quesiti esaminando con cura i libretti di istruzione delle apparecchiature, o del software, in loro possesso, gli arretrati di "Commoodre Computer Club" o i fascicoli nel frattempo apparsi in edicola. Alcuni dei lettori qui di seguito nominati troveranno opportuni chiarimenti nei fascicoli di imminente pubblicazione.

Christian Bonfanti - Arcore; Alfredo Bovenzi - Sessa Aurunca; Giulio Pagnossin - Venezia; Erich Zanellato - Genova; Valerio Capello - Nuoro; Gianfranco Venturelli - Salò; Fabrizio Conti - Roma; Roberto Mezzalira - Milano; Patrick Ghirardelli; Massimo P. - Santamarinella; Salvatore Caputo - Sciacca; Maurizio Ascione - Ercolano; Federico Cuciuc - Roma; Giuseppe Montrasio - Catania; Claudio; Ilario Benedetto; Antonio Bonadio - C. Calabro; Fabrizio Santinelli - Camucia; Salvatore Nivea - Malo.

AMIGA CONTRO MS-DOS

□ Viene spesso consigliato, anche sulla vostra rivista, il sistema Ms-Dos a coloro che chiedono consigli sulla necessità di usare macchine "professionali". Ma allora l'Amiga è eccezionale (come dicono) oppure no?
(Nunzio Santini - Comiso)

• La lunga lettera del nostro lettore (condensata per ovvi motivi di spazio) ripropone l'eterno dilemma di chi decide di abbandonare il C/64 e vuole, nel contempo, sbarcare verso altri lidi.

Il problema non è di semplice soluzione, nè si può risolvere citando un paio di battute, più o meno pertinenti.

C'è da dire che, di fatto, lo standard "professionale" (piaccia o no) è rappresentato da Ms-Dos che, a sua volta, dovrebbe esser soppiantato dal nuovo OS-2 (ma solo per i computer basati sui micro 286/386) come assicurano, mese dopo mese, da oltre due(!) anni.

L'obiezione del lettore è più che legittima e diventa addirittura imbarazzante quando egli dice:

"...insomma: un Amiga dotato di 1 Mega di Ram con hard disk e drive da 3.5 è paragonabile ad un analogo Ms-Dos di pari memoria, corredato di hard e floppy disk?"

In teoria dovrebbe esser la stessa cosa (Guru meditation a parte) ma, in pratica, non è così. Sarà questione di moda o, peggio, di mancanza di "aperture" mentali; certo è che la realtà è proprio quella che vediamo.

Intandiamoci: conosco Amighisti che non cambierebbero il proprio eleboratore con due fust... computer Ms-Dos; ma vi sono anche irremovibili Ms-Dossiani che considerano l'Amiga poco più di un giocattolo.

Mi permetto di ricordare, ad ogni buon conto, che sull'altra nostra rivista "Personal Computer" verrà frequentemente affrontato l'argomento relativo alle "comparazioni" tra computer, periferiche e programmi con specifico riferimento al settore professionale (o, se preferite, "evoluto").

Nel frattempo, credete a me, non c'è fretta. Se non siete fermamente convinti delle prerogative del vostro prossimo acquisto, riflettete ancora; e documentatevi.

Nel frattempo, infatti, i prezzi scendono...

LA VOCE DEI LETTORI

LAMENTELE

Commodore Computer Club si ostina a parlare ai principianti ed a rubare pagine; lo spazio dedicato al C/64 ed al C/128 è stato ridotto quasi del 50%. Ad esempio il N. 60 riportava, tra l'altro, recensioni di giochi arcaici, articoli per possessori di Amiga e C/64 dotato di drive 1541, programmi buoni per bimbi di nove mesi, utility l.m. di scarsa utilità pratica, un articolo sul Raster praticamente inutile, articoli per chi ha una stampante o il Gw-Basic. In conclusione, per noi possessori di C/64 (privi di drive e stampante) zero pagine.

(Marco S.)

AMIGA, DA 500 A 2000

□ In commercio vengono poste alcune schede che, dotate di diversi connettori, renderebbero l'Amiga 500 praticamente eguale al modello 2000. Sono affidabili?

(C. G. - Lecce)

• Naturalmente riteniamo che la domanda sia legittima per chi già possiede un A-500 e vuole ampliare le potenzialità del suo computer; in caso contrario avrebbe più senso, infatti, acquistare direttamente un modello 2000.

Tornando al quesito, dobbiamo renderci conto che non si possono fare miracoli.

Ciò significa semplicemente che non vi sono grosse difficoltà nell'estrarre un bus (connettore) di 100 pin dall'A-500 e collegarlo ad una piastra esterna, magari racchiusa in un mobiletto metallico, dotata di vari connettori.

Bisognerebbe sapere che cosa c'è in questi connettori...

Scherzi a parte, la piastra viene, di solito, venduta "vuota", vale a dire che è solo in grado di ospitare le varie schede Commodore (Hard Disk, Janus Xt/At, Drive da 3.5 oppure 5.25, eccetera), da acquistare a parte.

Indispensabile, invece, è sottolineare il fatto che l'Amiga-500, per come è fabbricato e venduto, non può assolutamente sobbarcarsi l'onere di alimentare (elettricamente, s'intende) tutte le schede che possono esser montate sull'accessorio. Ne consegue che è doveroso acquistare anche un idoneo alimentatore, in grado di erogare la quantità di corrente richiesta.

Ma, anche supponendo di voler acquistare tutto l'occorrente, il "sistema" è affidabile o no?

Come intuitivo non è possibile rispondere alla domanda; è infatti indispensabile provare l'accessorio citato ("carico", ovviamente, di ogni scheda possibile) per verificarne l'affidabilità o, più sempicemente, il corretto funzionamento.

Se il marchingegno dovesse capitarci in Redazione, lo proveremo adeguatamente.

LA VOCE DEI LETTORI

DROGA

I computer: usarli poco per volta va bene, imparare il loro linguaggio anche, ma non si deve rubare il tempo dei divertimenti normali dei bambini tenendoli incollati davanti al monitor perchè si rischia di farli diventare oggetti passivi senza più difese contro un mondo a loro sempre più ostile. I bambini sono diventati, infatti, il mercato più florido per gli spacciatori di droga perchè sono abituati ad abboccare facilmente ad ogni tipo di "trappola".

I bambini devono tornare ai loro giochi innocenti, non devono essere rimbecilliti da un videogame che crea una dipendenza paurosa in quanto nel video si scarica la propria insicurezza credendo di ottenere, in cambio, l'essenza della vita mentre non è altro che un'illusione.

E quando si capisce la trappola, e si resta delusi, i più scaricano la delusione buttandosi nuovamente nel gioco. Questo ultimo particolare può provocare danni devastanti alla personalità del bambino, fino a farlo diventare un insicuro e, di conseguenza, un potenziale drogato.

Tutto questo per dire che non bisogna abusare del computer, ma bisogna usarlo stando attenti a non essere usati. Quindi, attenzione a non superare certi limiti.

(Mimmo T.)

COCKTAIL DI APPARECCHI

□ Come posso collegare i miei C/64 e C/128, contemporaneamente, ad uno stesso drive e, magari, anche ad una stampante?

(G. Gigantelli - Roma)

• Sul retro del drive sono presenti due connettori seriali destinati, rispettivamente, al computer e ad altre periferiche (come, appunto, una stampante) da connettere "in cascata".

Abbiamo spesso detto, ma quasi sconsigliato, che è possibile usare i due connettori per collegare il drive a due computer diversi, purchè i collegamenti siano effettuati ad apparecchi spenti e, soprattutto, purchè acceda al drive UN SOLO computer alla volta.

Di solito non succedono inconvenienti (nè hardware nè software) ma siccome il collegamento citato non è riportato sulla manualistica "ufficiale" non ci assumia-

RINGRAZIA

LE AZIENDE

KODAK • BYK GULDEN EMOFORM • ALIVAR MOTTA • ENEL • RCA COLUMBIA PICTURES • SNAM GAS METANO • IL TELEGRAFO • NABISCO SAIWA CIPSTER • TRAINI E TORRESI TRASPORTI • MINOLTA COPIATRICI • CORSICA FERRIES - SARDINIA FERRIES • SIMCO DURANGO • CORRIERE DI PORDENONE • ISA - SPRAY PAN • PEG PEREGO • MIELE ELETTRODOMESTICI • FRANCESCO CONTI • WEIGHT WATCHERS • DE .CA. EDIZIONI • SCIARE • SYSTEM EDITORIALE • EDIZIONI UNIVERSO • FININVEST COMUNICAZIONI • SODALCO NUVENIA POCKET • DOLMA RAIDER • RACHELLI PASTICCERIA • PRIMUS MODA • DOCK MASTER • VMG JONATHAN • NTC THAT'S • DAVOLI LATTICINI • MINISTERO DELLA SANITÁ PER AIDS • DIOMEDE • VICK INT. OIL OF OLAZ • TONNO NOSTROMO • EZIO FIORI PUMA • BLOOMING CASUAL • OCCHIALI DILLIMAN • NUOVA POLTI VAPORELLA • POLENGHI LOMBARDO • ALSO ENERVIT • OSAMA SCRITTURA • RVR ELETTRONICA • IMPERIAL TV COLOR • AMSTRAD • ONOS HOLOWATCH • F.&P. RISO GALLO • RCS • IL GIORNO SAGISA

LE AGENZIE

YOUNG & RUBICAM • GRUPPO ESSEVI • TED BATES • RSCG • SAATCHI & SAATCHI • BTB • SPRING • CLIO ADVERTISING • M.F. MEDIA • PMS • MARKINGEGNO • KAPPADVERTISING • MARIO BELLI ASSOCIATI • McCANN ERIKSON • BHT • PLURAL • GRUPPO ODG • MEDIA TEAM • VEPROMARK • RIGHTMIX • TONIC • GRUPPO ETHOS • ELLECIZETA

PER AVER CREDUTO AL CIRCUITO
PIANIFICANDO LE PROPRIE CAMPAGNE PUBBLICITARIE NEL 1988/1989

CERTIFICAZIONE USCITE:

tramite il servizio della AGB Italia, Egimedia fornisce agli utenti di TOP ITALIA RADIO l'ufficiale certificazione delle uscite dei comunicati pubblicitari.

TARGET MIRATI:

il network TOP ITALIA RADIO, grazie al tipo di palinsesto unificato per 6 ore al giorno, dà la possibilità di posizionare le uscite su target specifici.

COPERTURA NAZIONALE:

grazie all'adesione di circa 200 radio provincia, TOP ITALIA RADIO INTEGRATO garantisce una omogenea illuminazione del territorio nazionale con ottime quote di penetrazione.

FLESSIBILITÁ GESTIONALE:

data la centralizzata produzione ed edizione del network, Egimedia offre le più ampie disponibilità a risolvere esigenze artistico-creative, tecniche, logistiche e commerciali ad ogni utente pubblicitario.

TOP ITALIA RADIO METTE COLORE E FANTASIA NELLA TUA VITA

EGIMEDIA s.r.l. - 20121 Milano - via della Spiga 1 - Tel. 02/798531 - 794592 - 798632 - Fax 780400

LA VOCE DEI LETTORI

ERRORI INTENZIONALI?

Per ottenere che il programma CLI-VOX (C.C.C. N. 52) funzionasse a dovere, ho dovuto apportare numerose modifiche. Non ho intenzione di polemizzare, ma anche altre volte ho riscontrato errori nei listati da voi pubblicati, come ad esempio nel programma "Amiga Reciter".

Del resto non è poi del tutto negativo che ci sia qualche errore: anzi, è uno stimolo ad applicarsi ed imparare sempre di più. Potrebbe essere una buona idea inserire intenzionalmente qualche errore nei listati e, qualche numero dopo, pubblicare le soluzioni, come se si trattasse di un quiz.

Propongo, inoltre, che siano pubblicate più recensioni di programmi commerciali, sviluppate ponendo a confronto package dello stesso tipo.
(Lucio M.)

mo responsabilità di sorta; possiamo solo dire che noi stessi, spesso, configuriamo due computer ed un drive nel modo appena descritto.

Tutto procede egualmente bene se, invece di un drive, si connette una stampante. Non abbiamo mai provato, invece, a mettere insieme quattro apparecchiature (due computer un drive ed una printer); in teoria dovrebbe ancora funzionare, ma chi può dirlo con certezza?...

ROUTINE DI TOMA

□ Ho utilizzato il dischetto "Commodore Speciale: Routine grafiche" per far funzionare il programma "Movie Maker" (C.C.C. N. 51). Compare, però, un Syntax Error in 1360 la cui origine non comprendo.
(Paolo Molina - R. Emilia)
(Gabriele Badalotti - Vigevano)

• L'errore è sicuramente da attribuire all'istruzione "Char" che non è stata aggiunta al set di istruzioni del package "Routine Grafiche".

Come è infatti scritto nel fascicolo "Commodore Speciale" (che consente di realizzare grafica tridimensionale in prospettiva), le varie routine presenti sul dischetto devono essere assemblate secondo un particolare procedimento (descritto in fondo al fascicolo stesso) affinchè funzionino tutte al momento della loro attivazione.

Il Syntax Error riscontrato è la segnalazione tipica di un comando che... manca ancora all'appello.

MOUSE E POTENZIOMETRI

□ Vorrei "emulare" il mouse del mio Amiga mediante due potenziometri; è possibile?
(Francesco Baldacci - Livorno)

• Il mouse dell'Amiga è un dispositivo opto-meccanico (= ottico e meccanico insieme), nel senso che la sfera presente al suo interno trasmette il proprio movimento di rotazione a tre cilindretti, visibili rimuovendo la sfera, (e posti a 120 gradi l'uno rispetto all'altro e a diretto contatto con la sfera) solidali con altrettante miniruote a raggi.

Quando il mouse (o meglio, la sfera contenuta al suo interno) viene mosso, le tre miniruote ruotano a diversa velocità e in senso orario o antiorario a seconda del movimento impresso al mouse. I "raggi" delle tre miniruote, durante la rotazione, interrompono un minuscolo raggio di luce (generato costantemente da un led); in base alle interruzioni, alla loro frequenza ed al loro numero, il computer è in grado di stabilire se e in che direzione è stato spostato l'accessorio.

Si tratta, dunque, di un dispositivo che è parente stretto del joystick (al cui interno sono presenti micro-interruttori) più che delle paddle (al cui interno sono presenti, invece, potenziometri).

Ne consegue che non è possibile simulare (facilmente) le funzioni di un mouse con due potenziometri sia perchè i dispositivi che il computer controlla sono tre (le tanto citate miniruote) sia perchè viene attivata un'elaborazione relativa al conteggio di "scatti" (le interruzioni dei fasci di luce) che lavorano secondo la logica digitale ("tutto o niente").

E' noto, infatti, che è possibile inserire un mouse al posto di un joystick, a patto, ovviamente, di tener conto delle diverse modalità di funzionamento.

AUTOSTART SU AMIGA

□ Alcuni programmi commercializzati funzionano correttamente solo se si introduce il dischetto quando compare la manina che richiede il Workbench. Perchè, invece, non funzionano se, caricato il Workbench originale, si tenta di lanciarli cliccando sulla loro icona?
(Christian Zanatta - Preganziol)

• Il motivo è forse da ricercare nella particolare protezione applicata al dischetto. Un'altra spiegazione potrebbe riguardare l'attivazione di altri files, presenti sul disco, ma sprovvisti di icona. Spieghiamoci meglio.

Un dischetto di Amiga, compresi quelli "originali" presenti nella confezione del computer, possono contenere molti altri file oltre a quelli "legati" alle varie icone che appaiono su video in ambiente Workbench. E' noto, infatti, che vi sono diverse subdirectory contenenti varie routine richiamabili da CLI (Command Line Interface).

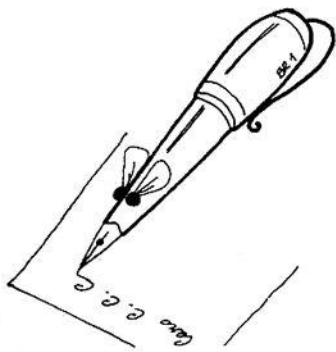
Tra i numerosi file ne può esser presente uno, in particolare, dal nome "Startup - sequence", che viene attivato tutte le volte che si accende il computer.

Il file in questione, in effetti, può essere a sua volta costituito da un "insieme" di comandi da eseguire secondo una particolare procedura.

Quando inserisci il disco, ed accendi il computer, viene quindi attivata automaticamente l'intera procedura necessaria a garantire il buon funzionamento del programma.

Caricando, invece, il "solito" Workbench e cliccando, poi, su una delle icone presenti sul disco interessato, si rischia di attivare solo alcune delle procedure stabilite, con comprensibili, successivi malfunzionamenti.

ARCHIVIO DISCO 128

□ Il programma che vi mando in visione mi ha impegnato per oltre 5 mesi; vorrei un giudizio sia sulla sua

FLOPPERIA®

presenta

in esclusiva per l'Italia

la **miglior** cartuccia mai prodotta per Commodore 64 e 128

MK V

Mk V' non solo è la miglior cartridge per effettuare copie di sicurezza del proprio software, ma è anche il più efficace velocizzatore nastro/disco e la più versatile cartuccia di utility esistente.
La sua peculiarità più innovativa è data dal suo **microprocessore interno**, appositamente studiato per sovrapporsi a quello del computer, ed assolutamente invisibile al sistema.
Ad esempio, mentre le altre cartucce si fermano ad un banale "Sprite Killer" per facilitare i giochi, Mk V' **è in grado di trovare automaticamente le "Poke" necessarie per le vite infinite** di qualsiasi programma presente e futuro, senza attendere che siano pubblicate dalle riviste o che qualche smanettone studi il programma. Ora anche voi potete produrre giochi **"trainer"**, senza alcuna conoscenza di linguaggio macchina! Inoltre **sprotegge e porta da nastro a disco (e viceversa) qualsiasi programma protetto, anche in multiload** (con i parametri in dotazione); può trasferire molti programmi e files dal formato 5"1/4 al nuovo **1581 da 3"1/2;** velocizza il nastro 5-6 volte oppure 8-10 volte, con velocità selezionabile; velocizza il disk drive come se fosse parallelo (**2 velocità: 202 blocchi in 9 secondi oppure in 6!**), ed è sempre efficace, anche con i programmi che disabilitano i fastload normali. Mk V' incorpora un vero e proprio editor di schermo, per poter cambiare più facilmente e velocemente le scritte nelle schermate o nei programmi; **funziona da interfaccia parallela**, per collegare una qualsiasi stampante standard Centronics al C64/128 e di usarla all'interno di qualsiasi programma, anche grafico; stampa o salva in qualsiasi momento la schermata o gli sprites di un gioco, per alterarli a piacimento. Aggiunge nuovi comandi al Basic, monitor L/M e disk, crea serie di immagini in sequenza su nastro, e tantissime altre cose ancora.

Per Commodore 64 e 128 (in modo 64), con qualsiasi registratore o disk drive, originali o compatibili.

In offerta a sole L. 105.000 (IVA compresa)

Manuale in italiano

Viale Monte Nero, 31
20135 Milano

Tel. (02) 55.18.04.84
Fax (02) 55.18.81.05

qualità sia sulla possibilità di commercializzazione dello stesso.
(Stefano Caranti - Ferrara)

• Il listato per C/128 cui si riferisce il lettore è un archivio per dischetti. E' sufficiente inserire, uno per uno, tutti i dischetti che un utente possiede per realizzare un data base che può facilitare la ricerca dei programmi memorizzati.

In linea di massima il programma, pur se discreto, è molto simile a tantissimi altri programmi analoghi che circolano liberamente tra gli appassionati.

La stessa procedura usata, velocità a parte, può essere realizzata con un buon programma di data-base.

Dubito che un software così particolare possa essere commercializzato, sia per la sua specificità, sia per il modesto numero di copie che è possibile vendere, sia per l'irrisolto problema della pirateria, che vanifica ogni sforzo.

Se lo ritieni opportuno potrò inserirlo su "Directory", ma non pensare di arricchirti: non è previsto compenso per tale collaborazione, a parte la soddisfazione morale di aver partecipato.

In quest'ultimo caso, inoltre, devi inviare il disco (senza protezioni) privo di autostart e, soprattutto, ridotto in un numero di blocchi più esiguo di quello che hai inviato: non posso occupare un intero disco con un solo programma!

LA VOCE DEI LETTORI

UTENTI MS-DOS, UDITE, UDITE!

La lettera di Licia P. di Firenze (vedi "La Posta" C.C.C. n. 60) che richiedeva maggiore spazio per i computer Ms-Dos, ha scatenato una vera e propria catena di S. Antonio: moltissime lettere e cartoline sono giunte in Redazione per sostenere, a furor di popolo, la tesi di Licia. Mai si era verificata una così vasta eco alla proposta di un lettore (pardon, lettrice) che ha creato un serio problema di organizzazione e di spazio.

Pur continuando ad affrontare su C.C.C, saltuariamente, argomenti legati al mondo Ms-Dos, abbiamo deciso di "trasferire" la parte più consistente degli articoli ad esso connessi sulla rivista "Personal Computer" che, affidata per l'occasione alla direzione del sottoscritto (il famigerato A. de Simone) dovrebbe accontentare la sete di notizie manifestata dai lettori interessati.

Un particolare ringraziamento, quindi, alla nostra gentilissima lettrice, alla quale va anche il merito di aver dimostrato che, spesso, una semplice lettera di "protesta" può portare frutti insperati.

Arrivederci, dunque, su "Personal Computer", la rivista degli smanettoni "evoluti" (e non).

RISPARMI ENERGETICI

□ Avete asserito che, durante il normale funzionamento del computer, è necessario tenere accese tutte le periferiche ad esso collegate. Non capisco, però, perchè devo tenere acceso inutilmente il drive se questo, una volta caricato un programma, non viene più adoperato.
(Cinzia Parenti - Roma)

• In passato abbiamo detto che, soprattutto nel caso in cui è collegata anche una stampante, da attivare nel corso dell'elaborazione, il drive DEVE restare acceso, anche se non serve; si verificano spesso, infatti, malfunzionamenti di vario tipo in caso contrario.

Ma non sempre è così. Molti programmi (tra cui giochi e/o utility) funzionano correttamente se la stampante viene tenuta spenta. In certi casi ancora, sebbene molto più rari, è possibile stampare solo se viene tenuto spento il drive durante la stampa.

Il comportamento "standard" rimane, tuttavia, unico: accendere tutte le periferiche PRIMA di accendere il computer, e lasciarle sempre sotto tensione. Solo in casi particolarissimi è possibile spegnere uno o più apparecchi.

Per ciò che riguarda il lodevole intento di evitare sprechi, mi permetto di ricordare che un drive consuma solo 25 watt, vale a dire la metà di ciò che consuma la lampadina di un abat-jour poco luminoso.

I risparmi di energia elettrica riguardano ben altre categorie di consumi.

AMIGA E L.M.

□ Come scrivere ed attivare, con Amiga, un programma in linguaggio macchina?
(Fabrizio Conti - Roma)

• Il linguaggio macchina del microprocessore 68000, montato su Amiga, è del tutto diverso da quello del 6510 (del C/64) sia per il set di istruzioni (decisamente più potente ed agevole da usare) sia per il modo stesso di funzionare.

Come si può intuire, il discorso è decisamente complesso e non può nemmeno essere accennato in queste pagine.

Tra breve, però...

AMIGA AMMALATA?

□ Facendo funzionare un programma contenuto su un dischetto è apparso un messaggio relativo alla presenza di un virus; subito dopo il drive ha iniziato a girare e, dopo una successiva verifica, il floppy è risultato totalmente inutilizzabile. Ho un sospetto: gli altri miei dischi possono essersi infettati?

(Fabio Franchello)

• Se, dopo l'apparizione del messaggio, non hai spento il computer, ma hai continuato a lavorare come se niente fosse accaduto, è molto probabile che i dischetti successivamente inseriti siano stati infettati.

Se, però, era attiva la tacca di protezione, non dovrebbe esser successo nulla. Per sicurezza ti consiglio di "trattare" i dischi in tuo possesso con una delle procedure descritte più di una volta sulle pagine di Commodore Computer Club.

In futuro, quindi, usa pure dischetti di provenienza sospetta ma, dopo averli adoperati, SPEGNI il computer e ATTENDI un paio di minuti prima di RIACCENDERLO. Ti sconsiglio, infatti, di "trattare" preventivamente i dischetti in questione perchè potrebbero non funzionare correttamente in seguito al trattamento.

Il consiglio di tenere SEMPRE attiva la tacca di protezione (almeno finchè è possibile) dovrebbe essere, ad ogni buon conto, una regola... di vita.

risposte rapide

NON PER PLUS/4
(Antonio Rossi - Città di Castello)

Non esiste un programma di word processor per Plus/4 che consenta di stampare tenendo conto della particolare interlinea esistente nei fogli di carta bollata.

1571
(Luigi Francavilla - Potenza)

Il drive 1571, uscito di produzione, può essere utilizzato con il C/128 in modo 64, dal momento che emula perfettamente le funzionalità del 1541.

ASCENSORE
(Lino Fiorizzo - S. Antimo)

Su un numero di "Directory" abbiamo pubblicato alcuni dei programmi da te richiesti.

SORTEGGIO
(Paolo Crucitti - R. Calabria)

Non ci è possibile scrivere (e pubblicare) un programma che consenta di sorteggiare in modo opportuno i "turni" di squadre di calcio di un ipotetico campionato. Se qualche lettore lo ha sviluppato, si faccia avanti!

OPEN
(Raffaele Libanori - Ferrara)

Il comando Open 2,2,2 apre un file verso la porta utente, o meglio verso l'interfaccia Rs-232. Successivi comandi di invio e/o ricezione dati, pertanto, vengono eseguiti anche se non è presente una periferica collegata alla porta utente. Ecco perchè il computer sembra bloccarsi: in quel momento sta trasmettendo dati (o sta cercando disperatamente di riceverli) anche se non ce ne accorgiamo.

MODIFICHE A EASY SCRIPT
(Marcello Artioli - Ferrara)

Le modifiche cui accenni nella lettera sono molto interessanti e ritengo che sia opportuno riparlarne.

DATASSETTE CON C/64 EXECUTIVE
(Antonio Breda - Borgaro)

Il C/64 modello "Executive" (con video e drive incorporati) non dispone del connettore per il datassette: non è quindi possibile collegare il registratore se non effettuando modifiche hardware. Non disponiamo di tale computer e, quindi, non possiamo aiutarti. Se qualche lettore ha realizzato la connessione, ce lo faccia sapere.

MA CHE COLPA ABBIAMO NOI?
(Andrea Bontempo - Narni)
(Franco Vigoni - Varese)

La Systems Editoriale è soltanto una casa editrice, per di più indipendente. La richiesta di sostituzione dei libretti di istruzione in inglese di Amiga deve essere inviata esclusivamente alla Commodore Italiana.

INTERESSANTE
(Amigo Simone)

L'articolo che intendi scrivere mi sembra idoneo per una pubblicazione. Ne possiamo riparlare.

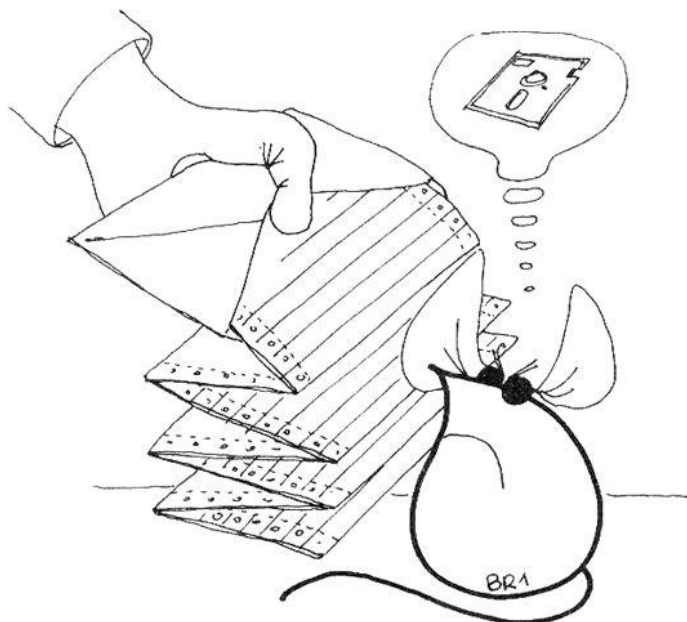
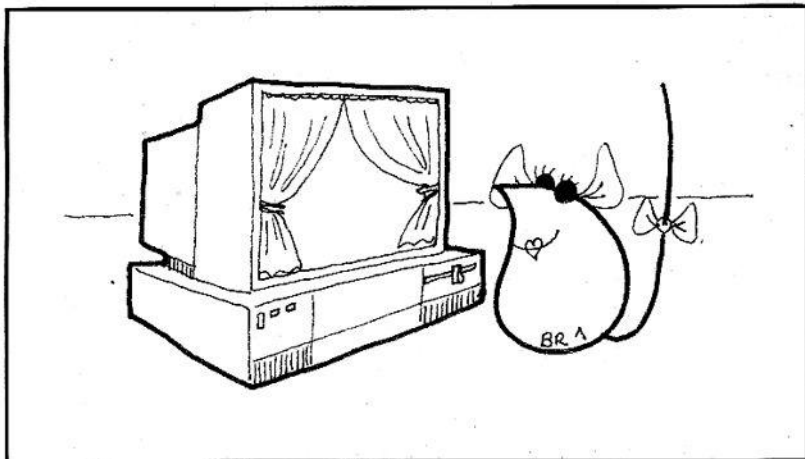
LIBRETTO ORIGINALE
(Giacomo Longo - Messina)

Nell'inserto speciale del N. 41, interamente dedicato al word processor Easy Script per C/64, sono stati riportati moltissimi dei comandi necessari per usare correttamente tale w/p, ma non tutti. Per conoscere l'intero elenco, e le funzioni che svolgono, è necessario procurarsi il libretto di istruzioni originale (costituito da parecchie decine di pagine).

SOLO SU DISCO
(Andrea Ponzio - Collegno)

Inviaci (solo su DISCO) il programma che, secondo te, presenta problemi: lo esamineremo con la massima attenzione e cercheremo di individuare l'errore.

ORRORI
(Paolo Molina - R. Emilia)

I comandi che usi per registrare e caricare file su/da disco è fondamentalmente errato. Ti consiglio di esaminare con cura sia il libretto di istruzione del drive, sia quello dell'Emulatore Gw-Basic per C/64.

UN MISCELATORE DI IMMAGINI

Vediamo in che modo agisce un genlock; e soprattutto a che serve!

Gli Àmici di Amiga

Nel 1986 uscì sul mercato americano il Genlock 1300, un dispositivo che permetteva di potenziare le capacità operative di Amiga, purtroppo funzionante solo con lo standard NTCS; finalmente oggi, a tre anni di distanza, la Newel vende un dispositivo simile come funzionamento, che può essere utilizzato in Europa essendo compatibile con le norme PAL.

Il Minigen, questo il nome della versione europea, è in grado di miscelare immagini, realizzate con Amiga (fisse o in movimento) con quelle provenienti da telecamere, videoregistratori, video-dischi o altri computer; si rende quindi più interessante una qualsiasi schermata video.

L'operazione effettuata dalla periferica di cui ci stiamo occupando è di "prendere", in entrata, l'immagine prodotta da Amiga, sostituirne il colore di sfondo con il segnale composito, coordinare i segnali di riferimento provenienti dalle due diverse fonti e "trasformarli" in un unico impulso video inviabile a televisori, monitor e videoregistratori che abbiano una presa di tipo CVBS o SCART.

Nella confezione è presente, oltre al minigen, un manuale che spiega in modo molto rapido, e senza spreco di parole, le modalità di installazione e le peculiarità del disposiivo; completa la dotazione un dischetto contenente un certo numero di video test, caricabili da Workbench, con i quali si può constatare il corretto funzionamento della periferica.

Non certo piacevole, anzi molto discutibile, è la totale assenza di cavi di collegamento che, nonostante siano facili da reperire o da realizzare in casa, rappresentano una difficoltà da parte dei non addetti ai lavori; era molto meglio fornire tutto l'occorrente, anche a costo di un leggero ritocco del prezzo di vendita.

La presa utilizzabile, lo riferiamo per dovere di cronaca, è una normalissima RCA di tipo maschio.

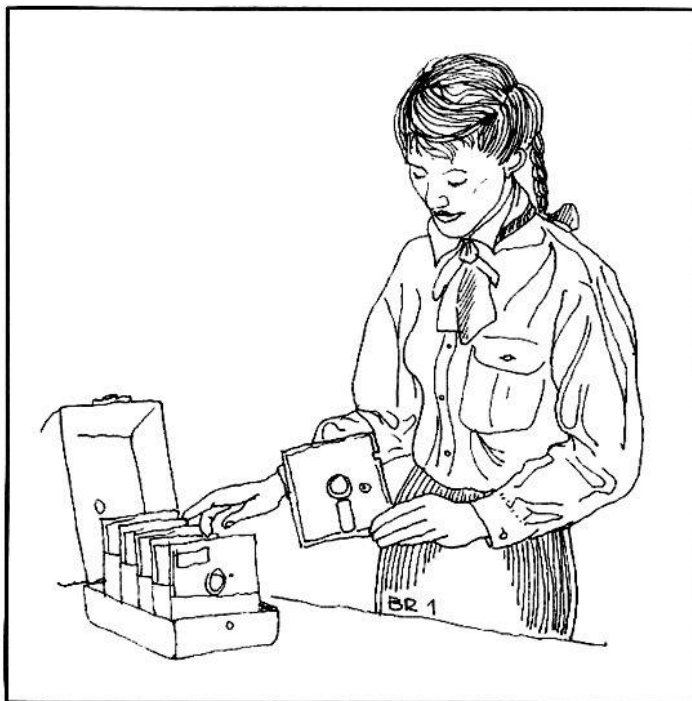
Il Minigen si collega direttamente all'uscita video di Amiga mediante lo stesso connettore dove, per intenderci, inserite normalmente il cavo di collegamento per il monitor; accetta solo segnali-video compositi ed è provvisto di un deviatore a tre posizioni che pilota i diversi modi di funzionamento:

Graphic.
Picture.
Mixed.

Il primo invia come uscita solo il segnale di Amiga comprendente anche il colore di sfondo; il secondo invia solo il segnale composito, presente in entrata; l'ultima modalità di funzionamento ha come output il video prodotto da Amiga con il colore di sfondo "rimpiazzato" dal segnale composito.

Al momento dell'accensione il computer, automaticamente, avvertirà la presenza del Minigen settandosi in modalità genlock; perciò non sarà necessaria nessun operazione da parte dell'utente se non quella di settare il deviatore sulla posizione desiderata. Da questo momento in poi la macchina opererà in maniera normale con tutti i programmi, senza aver problemi di incompatibilità.

Di solito il dispositivo viene utilizzato da alcuni network televisivi per far apparire messaggi in sovraimpressione o sequenze di immagini divertenti che possano intrattenere il pubblico; il suo utilizzo è anche adatto ad un hobbista, soprattutto grazie ai programmi appositamente realizzati che si possono acquistare presso qualsiasi rivenditore di software.

"Minigen" è un dispositivo veramente valido che, affiancato alle capacità straordinarie di Amiga, consente risultati stupefacenti, che non pensavate fossero possibili, e comunque in grado di soddisfare chiunque sia interessato al settore video.

MINI DESK TOP PUBLISHING

Mettere d'accordo testo e grafica è facile; vediamo come

di **Luca Morettoni**

Chiunque abbia esperienza nel campo giornalistico sa che il modo migliore per descrivere determinate informazioni (tra cui schemi e grafici) è il disegno.

I comuni word processor consentono di redarre buoni testi, ma non permettono di fondere insieme testo e grafica.

L'ostacolo sembra essere scavalcato, anche se solo in parte, dal GEOS.

Questo è senz'altro il miglior programma di D.P.T. reperibile in commercio per il C/64, ma è fonte di problemi per chi non ha mai avuto a che fare con operazioni di taglia - incolla e simili. Inoltre, per utilizzare il GEOS in modo "serio" occorre procurarsi un mouse.

Vedremo ora, al contrario, come possiamo, utilizzando il C/64, il drive, la stampante ed il simulatore GW-BASIC, realizzare dei documenti, più o meno validi esteticamente, infarciti di disegni realizzati, magari, con lo stesso GW-BASIC.

IL PROGRAMMA

Per stampare testi non vi sono assolutamente problemi (basta ricorrere intensivamente all'istruzione LPRINT); le difficoltà iniziano con la stampa dei grafici. Come fare? Basterà inserire nel testo opportuni codici in grado di ordinare al computer di caricare, e stampare, un certo grafico, ovviamente già presente su disco.

Il listato di queste pagine (contenente anche un text editor piuttosto rozzo, ma ampliabile) è in grado di accorgersi della presenza del "codice" di stampa (un asterisco, vedi riga 260); giunto a questo punto provvederà a caricare la schermata, dotata del nome presente dopo l'asterisco, ed a stamparla. Le schermate (che possono essere in numero qualsiasi) devono essere TUTTE presenti sullo stesso disco inserito nel drive durante la fase di stampa.

La fine del testo è indicata dal comando "-END-"

I TESTI

I testi possono essere scritti, come già detto, con l'editor interno (che permette anche di salvare e caricare file di testo), creati direttamente da DOS utilizzando in comando COPY CON (nella fase di stampa si possono utilizzare anche i file BATCH), oppure creati con un qualsiasi word-processor (EASY SCRIPT è l'ideale) e poi riconvertiti con il programma FCONV presente nel disco MS-DOS 2.

I GRAFICI

I grafici, invece, vanno salvati su disco con il comando BSAVE in quanto in fase di

stampa verranno caricati con il comando BLOAD. La sintassi completa del comando di salvataggio dei file grafici (che sono quelli presenti in SCREEN 1) è:
BSAVE "Nome file", 57344, 8000

E' molto importante indicare correttamente, nel testo, i nomi dei file grafici da utilizzare in fase di stampa in modo da evitare messaggi di "File not found" e successiva interruzione del programma.

TRUCCHI E MIGLIORIE

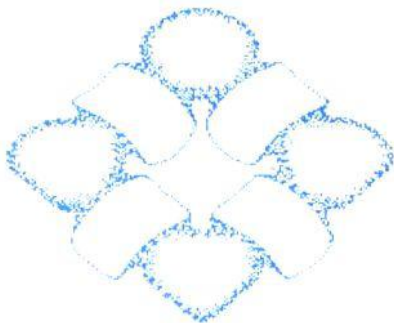
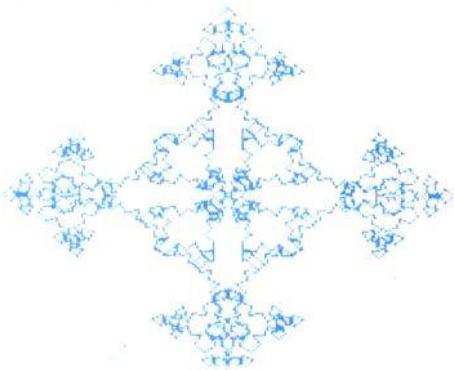
Non si può dire che il programma sia "pieno zeppo" di trucchi che richiedono periti in informatica (o volpini comuni) per essere compresi.

La fase che può sembrare meno chiara è quella di caricamento del file grafico. E' stato infatti utilizzato il buffer di tastiera (locazioni 631 - 640) per impartire il comando diretto BLOAD (righe 330 - 340).

Inoltre è presente un semplice BUFFER che si accerta che la schermata presente il memoria non sia uguale a quella da caricare: se ciò accade il file non verrà caricato, ma verrà subito stampato, con conseguente risparmio di tempo.

Tra le migliorie da apportare eccone alcune :

- "Tradurre" il programma per altri computer
- Dotare il programma di un migliore TEXT EDITOR
- Creare, a parte, un programma che "traduce" le schermate provenienti da altri programmi grafici in file compatibili.
- Possibilità di modificare il testo prima della stampa.

```
QUESTO PROGRAMMA PERMETTE DI STAMPARE UN QUALSIASI TESTO INSIEME A DEI
GRAFICI.
QUESTO E' IL PRIMO GRAFICO.
```

```
E' POSSIBILE SERVIRSI DI VARI TESTI GIA' CREATI CON IL COPY CON DEL DOS
OPPURE DI ALTRI WORD-PROCESSOR CONVERTITI CON L' UTILITY FCONV.
I GRAFICI INVECE DEVONO ESSERE PREPARATI DA GWBASIC USANDO LE VARIE ISTRUZIONI
GRAFICHE E POI SALVARLE SU DISCO CON
IL COMANDO BSAVE 'NOME'
IN UN TESTO POSSONO ESSERE INSERITI MOLTISSIMI GRAFICI.
```

```
IL TESTO DOVRA' TERMINARE CON IL COMANDO '-END-'
```

```
10 REM PROGRAMMA "MINI DESK TOP PUBLISHING"
20 REM APPLICAZIONE GRAFICA
30 REM BY LUCA MORETTONI
40 REM PER COMPUTER C/64 DOTATO DI GWBASIC
50 :
100 CLS:CLS 1:KEYOFF:COLOR 6,14,6
110 P$="":FOR A=1TO39:P$=P$+"-":NEXT
120 PRINTP$
130 PRINT" MINI D.T.P PER C64 (RICHIEDE GWBASIC)"
140 PRINTP$
150 DIMRG$(700):CG$="":CR=0:REM PREPARA LE VARIABILI
160 INPUT"VUOI CARICARE UN FILE DI TESTO ";A$
170 IFA$="S"THEN360
180 PRINTP$:PRINT"DIGITA IL TESTO , FINISCI CON -END-":PRINTP$
190 LINEINPUTRG$(CR):REM IMMISSIONE DEL TESTO
200 IFRG$(CR)="-END-"THEN230
210 IFCR=700THEN230
220 CR=CR+1:GOTO190
230 PRINTP$:INPUT"VUOI SALVARE IL TESTO ";A$:IFA$="S"THEN450
240 COLOR6,6,6:CLS:TD=0:REM INIZIA A STAMPARE
250 IFTD=CRTHENCOLOR6,14,6:CLS:END
260 IFMID$(RG$(TD),1,1)="*"THEN300:REM SE E' UNA SCHERMATA LA STAMPA
270 IFRG$(TD)=""THENLPRINT:GOTO290
280 LPRINTRG$(TD)
290 TD=TD+1:GOTO250
300 X$=MID$(RG$(TD),2,11)
310 IFCG$=""THEN330
320 IFX$=CG$THENLCOPY1:GOTO290:REM SE IL FILE E' IN MEM. LO STAMPA ...
330 CLS:LOCATE1,1:PRINT"BLOAD ";CHR$(34);X$;CHR$(34):LOCATE4,1:PRINT"GOTO350"
340 POKE631,19:POKE632,13:POKE633,13:POKE198,3:END:REM ALTRIMENTI LO CARICA
350 LCOPY1:CG$=X$:GOTO290
360 REM CARICAMENTO DEI DATI
370 PRINTP$:INPUT"FILE NAME ";F$
380 IFF$=""THEN180
390 IFF$="$"THENCLS:FILES:F$="":GOTO370
400 OPEN "I",#1,F$:CR=0
410 IFEOF(1)=1THEN440:REM SE IL FILE E' FINITO ESCE
420 CR=CR+1
430 INPUT#1,RG$(CR):GOTO410
440 CLOSE 1:GOTO240
450 REM SALVATAGGIO DEI DATI
460 PRINTP$:INPUT"FILE NAME ";F$
470 IFF$=""THEN240
480 IFF$="$"THENCLS:FILES:F$="":GOTO460
490 OPEN "O",#1,F$:C=0
500 IFC-1=CRTHEN530
510 PRINT#1,RG$(C)
520 C=C+1:GOTO500
530 CLOSE1:GOTO240
540 END
```

```
questo programma permette di stampare un qualsiasi testo insieme a dei
grafici.
questo e' il primo grafico.
*graph-1.pic
e' possibile servirsi di vari testi gia' creati con il copy con del dos
oppure di altri word-processor convertiti con l' utility fconv.
i grafici invece devono essere preparati da gwbasic usando le varie istruzioni
grafiche e poi salvarle su disco con
il comando bsave'nome',57344,8000
in un testo possono essere inseriti moltissimi grafici.
*graph-2.pic
il testo dovra' terminare con il comando '-end-'
-end-

-----------------------------------
```

Esempio di testo memorizzato con DTP/64. Si noti la particolare sintassi con cui indicare il caricamento di schermate grafiche (per la successiva stampa).

STRAZIAMI, MA DI BATCH SAZIAMI

Un po' di pratica con i potenti e versatili comandi di Amiga

Gli amici di Amiga

Un file batch, come (dovrebbe esser) noto, non è altro che una sequenza di istruzioni, memorizzata appunto in un file, che viene eseguita dal computer allorchè viene richiamata dall'istruzione Execute.

In pratica, e con versatilità molto superiore, è come se digitassimo direttamente da CLI una lunga serie di comandi del Dos, premendo Return dopo ognuno di essi; oppure ancora, più similmente, inserendo un "Control - J" tra l'uno e l'altro prima del Return.

Se, per esempio, si digita...

echo "questa è una directory"

...e si preme Control+J, si digita ancora Dir e, finalmente, Return, si vedrà dapprima apparire la stringa ("questa è..."), seguita dall'elenco dei programmi presenti nella directory corrente (esecuzione vera e propria di DIR).

Come è intuibile, seppur comoda, l'opportunità di usare "Control J" nel modo descritto non consente poi grosse realizzazioni e rimane come una semplice curiosità.

AMIGA, PRIMI PASSI

Il primo approccio con un gioiello come l'Amiga, invariabilmente, è legato alle comode e rilassanti escursioni del mouse tra le altrettanto comode, e graficamente impeccabili, finestre del Workbench.

Qualche "click" ben assestato, e l'intero universo informatico sembra improvvisamente a portata di mano.

Per molti il discorso finisce lì, e in fondo non c'è niente di male: di software dedicato, ormai, se ne trova a bizzeffe, e continua ad esserne sfornato altro con rassicurante regolarità.

C'è, però, anche chi ha definitivamente aperto un'altra finestra, quella del CLI, scoprendo il piacere di dialogare più da vicino con una macchina così potente, assoggettandola (si fa per dire!) ai propri voleri con ben maggiore incisività che con il Workbench.

Il rovescio della medaglia è certamente noto, specialmente ai novizi: la digitazione dei vari comandi del Dos, almeno fino alla diffusione della nuova Shell versione 1.3, è a dir poco da... ulcera.

Un minimo errore, e bisogna riscrivere tutto, per non parlare del labirinto di directory, assegnazioni e path vari in cui ci si perde, allorchè si prova a mettere in atto qualcosa di più complesso di un semplice List.

Per il primo inconveniente, come si è già detto, non resta che affidarsi alla versione 1.3, "ufficiale" o meno che sia.

Per il resto, ci viene incontro lo stesso Amiga, o meglio il suo Dos, con la possibilità di far eseguire un blocco di comandi predefiniti, i cosiddetti batch file, strutturabili in modo da poter tranquillamente essere identificati come veri e propri "programmi".

La nostra rivista se ne è già occupata sul n. 47 (Nov. '87) e (limitatamente al sistema MS-DOS), sul n.60, per cui ci limiteremo, in questa sede, a pochi accenni teorici, soffermandoci con più attenzione su possibili realizzazioni pratiche.

Ciò, tra l'altro, consentirà anche ai meno esperti di (ri)avvicinarsi ad alcuni dei meno noti comandi dell'Amiga Dos.

Per disporre di un batch file occorre anzitutto prepararlo, adoperando, come già descritto sul n.47, l'editor ED contenuto nel disco di sistema, o, in alternativa, Edit (contenuto nello stesso dischetto).

E' anche possibile usare un qualunque word processor in grado di salvare il solo testo in formato ASCII, privo, cioè, dei caratteri di controllo del programma elaboratore di testi (tipo TextCraft o Prowrite); esiste, tuttavia, una soluzione alternativa, forse più semplice.

Non tutti, infatti, hanno dimestichezza con l'enorme, e poco convenzionale, quantità di comandi disponibili con ED o Edit, e la pigrizia può scoraggiare l'uso di un word processor.

Ebbene, per coloro che rientrano in queste categorie, è sufficiente usare... il CLI stesso.

Basta, infatti, digitare..

*Copy * to nomefile*

...e stilare il testo con i vari comandi, avendo l'accortezza di premere il Return dopo ogni riga.

Ad ogni ritorno di carrello la riga presente sul rigo stesso verrà memorizzata su dischetto (nella directory corrente, se non si specifica diversamente) nel file con il nome indicato ("nomefile", nel poco fantasioso esempio appena riportato), "accodandosi" alle altre righe già presenti, fino a che non verrà premuto Control + back slash (barra inversa).

Qualora si decidesse per questo metodo, occorre prestare attenzione a quanto si è scritto, prima di premere il Return: "dopo", infatti, non si ha più la possibilità di correzioni, a meno di ricorrere a veri (e scomodi) editor di testo.

Quale che sia il metodo utilizzato, il luogo di elezione per i file-comando è la directory S (= Sequence) del dischetto.

In effetti l'unico file batch che deve obbligatoriamente risiedere in questa directory è la startup-sequence, che viene letta ed eseguita dal sistema se un dischetto è stato reso "bootabile" con il comando Install.

Gli altri file possono trovarsi ovunque, purchè, per lanciarli, dopo Execute venga precisata la loro posizione.

Se, per esempio, abbiamo memorizzato un file con nome "cavoli" nella directory "verdure" del dischetto "vegetali" (floppy erborista?), questo andrà attivato con...

Execute vegetali: verdure / cavoli

Se, invece, il file "cavoli" si trovasse nella directory S, basterebbe un...

Execute cavoli

...per essere rintracciato automaticamente.

LA PAROLA AI FATTI

Una doverosa precisazione riguarda quelli che, genericamente, abbiamo finora definito con il termine "comandi" del Dos.

In pratica si tratta dei vari programmi presenti nella directory C del dischetto di sistema, liberamente inseribili nel contesto di un batch file.

E' chiaro che, volendo editare un file comando in un altro dischetto "indipendente", questo deve contenere, preferibilmente nella sua directory C, tutti i comandi presenti nel batch file.

Si tenga presente, inoltre, che il numero dei comandi disponibili è praticamente infinito: nulla vieta, infatti, di aggiungere nella directory C altri programmi che svolgono le più disparate funzioni, richiamabili, a parte, con il nome loro assegnato o da un nostro batch file o direttamente da tastiera.

Un esempio: in uno dei dischetti "Amigazzetta" è presente un programma, di nome Show, che serve per mostrare, direttamente su video, schermate archiviate su disco in formato IFF.

Se si copia il programma nella directory C del disco, ove verrà anche memorizzato un nostro batch file, si potrà adoperare una riga tipo...

Show nomedisegno

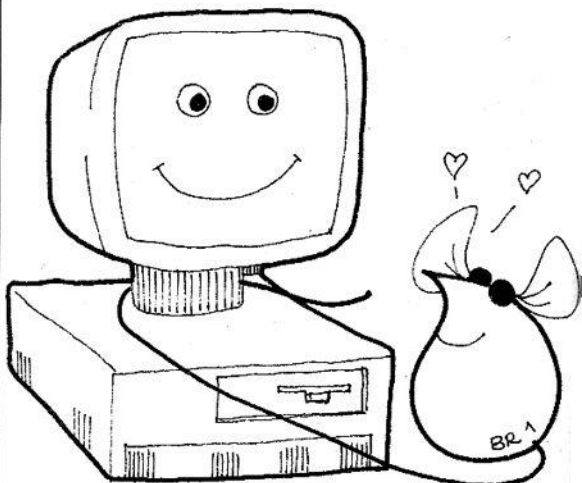
...che svolgerà correttamente il suo compito.

Ancora meglio, si potrebbe inserire questa istruzione all'interno della startup-sequence, in modo da far apparire la schermata al momento del boot del disco!

Ma vediamo di adoperare concretamente i comandi "ortodossi" (quelli presenti nel disco Workbench 1.2), cominciando proprio dalla startup-sequence, che, lo ribadiamo, non è altro che un batch file.

PARTENZE INTELLIGENTI

Supponiamo che in un dischetto, fornito di autostart (in caso contrario, si adoperi "In-

```
;  ------------------------------------
;                 FILE 1
;  ------------------------------------
ask ,"Workbench? (y/n)"
if not warn
  if exists :docs
    list :docs
  endif
  echo " "
  echo "Premi return per"
  echo "caricare notepad"
  ask^>nil:
  run notepad
  quit
endif
if exists :disk.info
  copy :disk.info to ram:disk.info
endif
loadwb
endcli
```

stall nomedisco:"), siano presenti tre programmi che usiamo frequentemente, e che si voglia disporre di un menu per accedere rapidamente al programma desiderato.

Ebbene, niente di più facile: basterà prima creare un file di nome Menu (o altro a scelta), con una delle tecniche di editing già accenate, più o meno così strutturato...

1) Nome programma 1
2) Nome programma 2
3) Nome programma 3

...e memorizzarlo nella directory principale del dischetto.

Quindi, assegnare (con Rename) al primo programma il nome "1", al secondo "2", ed al terzo "3".

A questo punto basterà editare, e salvare nella directory S, questo file di nome "startup-sequence"...

type menu
prompt "SCELTA >"

...ed il gioco è fatto.

In pratica, dopo un autostart del disco, prima verrà stampato sullo schermo il contenuto del file Menu, dopodichè, restando in CLI, viene solo modificato il prompt "1>33 che normalmente appare.

Premendo uno dei tre tasti (1, 2, oppure 3), e subito dopo Return, verrà (come di norma in ambiente CLI) lanciato il programma corrispondente.

Sistema grossolano quanto si voglia... ma efficace.

Data, però, la disponibilità di strutture come If...endif, si può aspirare ad applicazioni ben più sofisticate.

Vediamone un esempio, riferendoci stavolta al File n. 1 illustrato in queste pagine.

Si tratta di un'altra startup-sequence, mirata ad un eventuale dischetto di lavoro basato sull'uso del Notepad, l'editor di testi fornito gratuitamente con il Workbench.

Nulla vieta, ovviamente, che al posto di Notepad venga inserita una qualsiasi altra utility di scrittura, ed in ogni caso il batch file proposto deve essere inteso come una base da ampliare e personalizzare a vostro uso e consumo.

Nel nostro esempio, dopo lo start del disco, il file chiede se si desidera entrare in Workbench o meno.

L'uso di "ask", poco conosciuto, è molto semplice: digitando Y, viene soddisfatta la condizione "Warn", altrimenti no.

In ogni caso, viene stampato il messaggio che segue Ask (le virgolette si usano solo se sono presenti degli spazi), ed il sistema resta in attesa della pressione del tasto Return.

Nel batch file, quindi, tramite un primo ciclo If...endif, se non si è risposto positivamente (If not warn = opposto di If warn) si procede; altrimenti entrano in azione i comandi posti dopo Endif, che caricano il Workbench e chiudono la finestra CLI.

Diversamente, prima viene listata una eventale directory di nome docs (nome facoltativo), ma solo se è presente (=If exists), e quindi "nidificando" nel precedente If...Endif una seconda struttura analoga.

La successiva richiesta di pressione di Return è inserita, più che altro, a scopo dimostrativo, per illustrare come Ask possa essere sfruttato in modo anche più banale.

In questo caso, con >nil: si evita la comparsa, sullo schermo, di eventuali caratteri poco eleganti.

E finalmente, sempre se si è risposto "N" alla prima richiesta, viene lanciato il Notepad con Run, in modo da lasciare utilizzabile la finestra CLI, ovvero la task 1.

Obbligatorio risulta il "Quit" finale, altrimenti verrebbero eseguite le istruzioni presenti successivamente.

Nel caso si opti per il Workbench, e sempre subordinatamente alla condizione Exists, si copia il file Disk.info nella Ram Disk.

Il motivo è puramente estetico: l'icona-disco può essere stata modificata con l'Icon Editor, o semplicemente copiando da qualche altro disco in commercio (magari col famoso Climate...) il file disk.info.

In tal caso, anche l'icona del Ram Disk modificherà il suo solito (e tetro) aspetto sullo schermo del Workbench.

CAMBIO INDOLORE

Abbandoniamo ora le sequenze di start, e riserviamo la nostra attenzione alle interessanti modalità di Execute.

Il dato più importante da rilevare è la possibilità di inviare parametri al file batch, semplicemente accodandoli (dopo uno spazio) al nome.

In pratica, sono possibili due sintassi:

Execute Nomefile (già accennata in precedenza)
Execute Nomefile Parametro1 Parametro2 (eccetera)

Nel secondo caso, il batch file dovrà contenere una direttiva .Key, come risulterà più chiaro esaminando il File 2 di queste stesse pagine, che può risultare utile in più di una occasione.

Vi sarà certo capitato, dopo aver iniziato una sessione di lavoro, di dover operare massicciamente su un dischetto che non sia quello adoperato per il boot, tipicamente il disco Workbench.

In questo caso, anche chi possiede il drive esterno (pressochè obbligatorio!), spesso si trova ad obbedire alle continue richieste di reinserire il disco di sistema.

Unica alternativa, riassegnare (con Assign) tutti i Device logici, vale a dire *C:, S:, L:, Libs:, Devs:, Sys: e Fonts:.*

Più praticamente, possiamo però affidare il compito ad un file-comando come quello proposto, memorizzato nella directory S, limitandoci ad impartire...

Execute nomefilebatch nomenuovodisco

...senza il carattere di doppio punto (:) finale.

Come si può notare dal listato, la prima direttiva è proprio .Key, seguita dal parametro arbitrariamente definito "disco".

Se i parametri fossero stati più di uno, si sarebbero dovuti separare tra di loro con una virgola, e senza eccedere il limite di 255 caratteri.

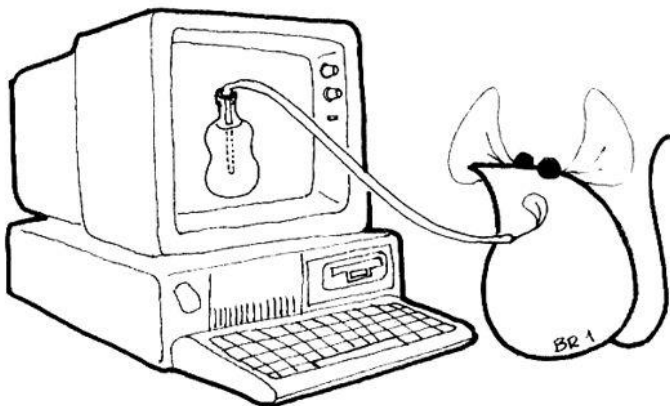
Ebbene, ogni volta che nel file verrà inserito il termine <disco>, racchiuso tra i simboli minore (<) e maggiore (>), questo verrà sostituito con il parametro passato per mezzo di Execute.

Il batch file in esame, abbastanza facile da comprendere, necessita solo di qualche precisazione sulle prime istruzioni, che controllano se sia inserito il nome del disco da assegnare come nuovo disco-sistema.

Per far ciò si adopera la forma If...Eq, che va interpretata così: se il parametro <disco> è nullo, salta alla subroutine Aborted.

Il problema è che Amiga Dos non riconosce le classiche doppie virgolette per indicare un parametro inesistente.

Ecco, allora, che si ricorre al trucco di aggiungere una X al nome del disco, per poi paragonarlo alla sola X, indicante che il parametro... non c'è.

Il salto successivo, come dovrebbe essere noto, è specificato da...

CACCIA AL FILE

Il Dos dell'Amiga, come ogni altro Dos, si occupa della gestione di periferiche hardware come i disk drive, o la stampante, ma non solo di queste.

Di notevole importanza per il sistema, nonchè per chi lo sfrutta, sono anche le cosiddette Device (=periferiche) logiche, tipicamente corrispondenti a delle directory del disco di sistema.

Chiariamo il concetto.

Il sistema operativo di Amiga è strutturato in modo da non risiedere totalmente in memoria: solo al momento del bisogno utilizza vari files presenti sul dischetto, cercandoli nella directory corrente (quella specificata dal comando Cd).

Se non li trova lì, ecco che entrano in ballo le device logiche che, in pratica, "suggeriscono" al sistema dove andarli a cercare.

Sappiamo, per esempio, che i comandi del Dos sono contenuti nella directory C (=command) del disco Workbench.

Deve esistere, dunque, una Device logica C: (i due punti servono a specificare che si tratta di un device, e non di una directory), che dirà al sistema operativo: "se non trovi il comando nella directory corrente, vallo a cercare nella directory C".

Digitando, da CLI, il comando Assign, verrà mostrato l'elenco delle device logiche attive, seguito dalle periferiche fisiche.

Si noti come, ad ogni Device, corrisponde una directory nel disco Workbench, con SYS: che sta ad indicare quella principale (root, cioè "radice").

Sempre tramite il comando Assign, possiamo modificare questi "promemoria" del sistema operativo, indicando percorsi alternativi a noi più utili, come descritto nel presente articolo.

Ecco che, ad esempio, nel file 3, con Assign C: Ram:, si farà in modo che i comandi vengano cercati non più nella directory C del disco, ma nella directory principale della Ram Disk, mentre nel file 2 tutte le assegnazioni vengono dirottate verso il nuovo disco sistema (nelle corrispondenti directory, se esistono).

Oltre che modificare le device logiche, è possibile anche aggiungerne di nuove.

Sempre riferendosi al file 3, con Assign X: Ram:copy, si fa in modo che, ogni qualvolta venga invocata la Device X:, il sistema esegua il comando copy presente in Ram, con notevole guadagno in termini di velocità.

Limitandosi a citare la Device dopo Assign, la stessa verrà eliminata.

Le modifiche apportate alla definizione dei Device logici, possono sempre essere visualizzate con Assign, che "aggiorna" l'elenco in funzione dei nostri interventi.

```
; -----------------------------------
;                FILE 2
; -----------------------------------
.key disco
if <disco>x eq "x"
  skip aborted
endif
if not exists <disco>:
  quit
endif
cd <disco>:
if exists <disco>:l
  assign l: <disco>:l
  else
  echo "<disco>:l non assegnata"
endif
if exists <disco>:libs
  assign libs: <disco>:libs
  else
  echo "<disco>:libs non assegnata"
endif
if exists <disco>:devs
  assign devs: <disco>:devs
  else
  echo "<disco>:devs non assegnata"
endif
if exists <disco>:fonts
  assign fonts: <disco>:fonts
  else
  echo "<disco>:fonts non assegnata"
endif
if exists <disco>:s
  assign s: <disco>:s
  else
  echo "<disco>:s non assegnata
endif
if exists <disco>:c
  assign c: <disco>:c
  else
  echo "<disco>:c non assegnata"
endif
assign sys: <disco>:
assign
quit
;
lab aborted
echo "manca il nome del disco!"
```

Skip nomesubroutine

...che dirigerà il flusso delle operazioni al punto marcato dalla direttiva Lab nomesubroutine.

Anche qui il salto non sarebbe strettamente necessario (la didattica vuole la sua parte!): si potrebbe inserire la Echo della routine Aborted direttamente dopo la condizione If, aggiungendo Quit prima di Endif.

Inoltre, piuttosto che il nostro controllo, potrebbe essere adoperata la forma...

.Key disco/a,

```
;----------------------------
;           FILE 3.
;----------------------------
assign y: sys:c
copy y:copy to ram:copy
copy y:assign to ram:assign
assign c: ram:
assign x: ram:copy
x: y:if c:if
x: y:endif c:endif
x: y:else c:else
x: y:skip c:skip
x: y:echo c:echo
x: y:quit c:quit
x: y:cd c:cd
x: y:execute c:execute
; x: s/file2 c:file2
assign x:
assign y:
cd c:
```

... che rende obbligatoria la presenza del parametro dopo Execute.

In sua assenza, la procedura si arresta con un sibillino (Parameters unsuitable for key "nomedisco/a"), decisamente meno chiaro del nostro.

Fate un po' voi.

Il batch file proposto, così com'è, evita una discreta mole di lavoro, ma solo se si possiede un secondo drive.

In caso contrario, pur funzionando a dovere, rischia di provocare seri... crampi alla mano destra, per il gran numero di cambi di disco richiesti.

Anche per i malcapitati, comunque, un rimedio c'è: usare la Ram Disk.

In pratica si tratta di copiare, e mandare in esecuzione, prima del File 2 appena visto, il File 3.

In alternativa i due file possono essere compresi in uno unico, inserendo, per primo, il file 3.

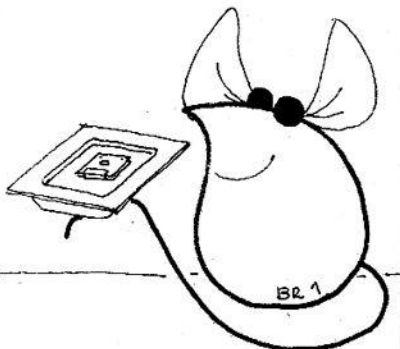
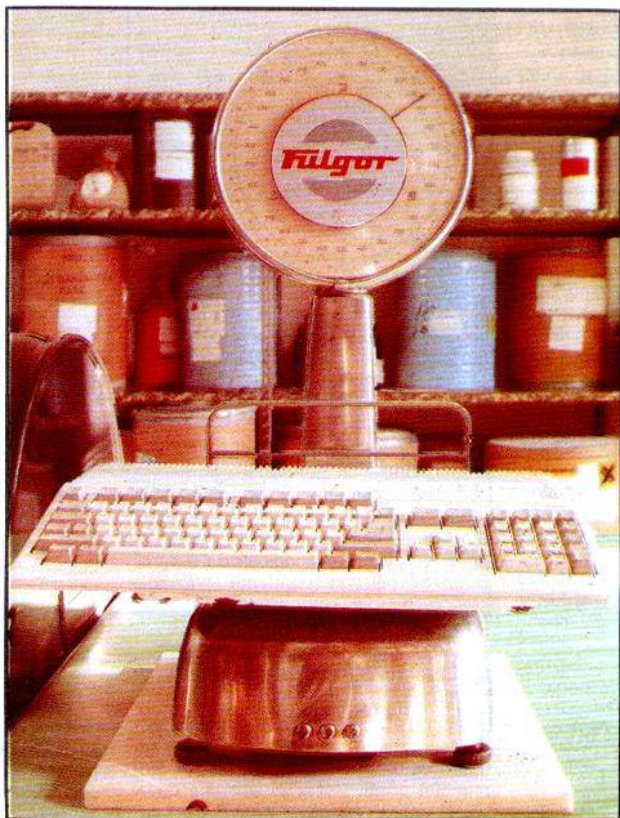
Questo, molto semplicemente, copia in Ram Disk tutti i comandi presenti nel file 2, che non saranno così più cercati nel disco originario, assegnando provvisoriamente al device Ram: le veci del device C: (vedi riquadro).

Si noti che, preceduto da un punto e virgola (e quindi non utilizzato), è presente anche un comando che copia in Ram il File 2.

Qualora si desiderasse avere due file batch (il "file2" ed il "file3") piuttosto che uno unico, il punto e virgola andrebbe eliminato.

Come risultato di queste ultime operazioni, non sarà necessario cambiare continuamente dischetto, e la tecnica può essere applicata anche per altri scopi.

Un'ultima nota: a lavoro concluso, se il nuovo disco-sistema è protetto in scrittura, può apparire una segnalazione del sistema per avvertirci del fatto.

Ci si limiti a "clickare" su Cancel, è solo un tentativo del Dos di registrare un file di backup, peraltro senza altre implicazioni.

ALLA SCOPERTA DEL PC NASCOSTO NELL'AMIGA

Forse non tutti sanno che Amiga in grado di trasformarsi in un piccolo "clone" Ibm senza (o quasi) nessuna spesa aggiuntiva

di **Michele Maggi**

E' ormai chiaro che l'Amiga destinato a prendere il posto del buon vecchio C/64 anche se quest'ultimo si difende ancora egregiamente.

"Purtroppo", per, l'evoluzione tecnologica non lascia scampo: le prestazioni aumentano e, con loro, anche le aspettative da parte dell'ormai esigente clientela.

In questo contesto, specialmente se ci riferiamo al settore "Hobby & Home computer", il fenomeno Amiga non può assolutamente essere ignorato: che piaccia o no, il nuovo computer della Commodore è una realtà assolutamente non trascurabile, soprattutto tenendo conto del prezzo di vendita, ormai paragonabile a quello del più modesto C/64.

Apparentemente la macchina sembra presentare ostacoli per ciò che riguarda la comprensione del suo "intimo" funzionamento. Con un po' di pazienza, tuttavia (e facendo ricorso a un discreto numero di manuali tecnici "riservati"), non è difficile muoversi agevolmente fra Copper, Paula e Blitter.

Stavolta, però, parleremo di un dischetto che, nonostante circoli da parecchio tempo, viene spesso disdegnato da parte degli "addetti ai lavori".

Il software in questione è il "Turbo Tran-

TABELLA RIASSUNTIVA BENCHMARK AMIGA/AMSTRAD		
	Amiga + Transformer	Amstrad PPC 640
TESTO:	20 min. 58 sec.	1 min. 17 sec.
SCRITTURA FILE:	12 min. 10 sec.	57 sec.
LETTURA FILE:	10 min. 59 sec.	55 sec.

La tabella riassuntiva dei risultati del benchmark

former V1.2" che tutti gli utenti Amiga magari possiedono, senza saperlo, fra la sterminata biblioteca software (pirata?) che correda il calcolatore.

La validità del software di cui ci occupiamo è davvero notevole: consente infatti di trasformare il fedele Amiga in un PC Ibm compatibile (o quasi).

Una volta caricato il software (clickando sull'icona raffigurante la bombetta e il bastone, tipico simbolo Ibm) appare una schermata con un messaggio che richiede l'inserimento di un dischetto di sistema MS-Dos.

A questo punto la maggior parte di coloro che hanno provato ad utilizzare Turbo Transformer avranno desistito dal continuare a causa della mancanza del dischetto in questione.

Peccato! Varrebbe proprio la pena di procurarsene una copia (da un amico o da un collega o, magari dal vicino di casa) per fare qualche prova.

UN DOS PER IL MIO AMIGA

Per ciò che ci riguarda, abbiamo reperito un dischetto di sistema di un computer portatile Amstrad (modello PPC 640-D).

Dopo un discreto periodo di tempo trascorso per il caricamento dei file di sistema, abbiamo finalmente avuto il piacere di vedere l'agognato prompt "A>", tipico dei sistemi MS-Dos compatibili.

Digitata una prevedibile sequenza di comandi (DIR, CHKDSK, EDLIN, ecc.), abbiamo constatato il corretto funzionamento di tutti i comandi impartiti.

Ma, si sa, l'appetito vien mangiando, per cui il piccolo "compatibile" è stato sottoposto a qualche prova più sostanziosa.

LA PROVA PRATICA

Una prova realmente completa, ovviamente, dovrebbe essere estesa ad un numero rilevante di pacchetti Ms-Dos che possano dare un'idea di ci che su Amiga funziona oppure no.

La prima prova da noi stessi effettuata si è svolta usando WordStar 4.0, uno dei pi noti Word Processor per sistemi MS-Dos; contrariamente a quanto ci aspettavamo (noi della Redazione siamo sempre un po' cattivi...) il funzionamento stato ineccepibile, con l'unica "piccola" differenza: la velocità di esecuzione.

A causa dell'interfaccia software, infatti, l'esecuzione di un qualsiasi programma (o semplice comando) risulta rallentata in modo enorme in quanto "Turbo Transformer" deve necessariamente convertire ogni singola istruzione originale del sistema 8088/86 nel corrispondente linguaggio compreso dal processore Motorola 68000, CPU dell'Amiga.

In altri termini: la velocità di elaborazione circa 15/20 volte inferiore a quella di un vero PC.

A parte il "dettaglio" relativo alla velocità di elaborazione (non ci si può aspettare molto da un'emulazione software) è comunque positivo che un software complesso come WordStar non abbia causato problemi.

Un curioso inconveniente è invece rappresentato dal malfunzionamento dello SHIFT destro; per utilizzare le maiuscole infatti tassativamente necessario ricorrere allo SHIFT sinistro.

Immediatamente dopo WordStar abbiamo provato ad usare il compilatore QuickBasic 4.0 della Microsoft; questo ha fatto finta di funzionare (fino al momento della compilazione); poi ha preferito mandare l'Amiga in Guru Meditation.

Pazienza, ricarichiamo tutto da capo; Transformer e Dos.

La terza prova è stata effettuata tramite PC Tools, un software della Central Point, che svolge una quantità di funzioni particolari tra cui l'elenco delle caratteristiche della macchina su cui gira.

Il responso pubblicato in un apposito riquadro e vale la pena di esaminarlo per trarre qualche conclusione.

In primo luogo il processore (CPU TYPE) è catalogato come NEC V20; in secondo luogo (nota assai più dolente) la velocità calcolata rispetto al PC standard è stata valutata intorno al 10%

Se si considera che tutti i PC XT compatibili, basati quindi su processori Intel 8086, funzionano generalmente a 8 Mhz (200% rispetto al PC standard) la valutazione è alquanto bassa ma d'altronde, come gi sottolineato, non ci si possono aspettare miracoli.

Un'altra caratteristica non entusiasmante deriva dalla "scoperta" che il Transformer considera solo una faccia del dischetto da 3.5 pollici su cui lavora. In pratica, al posto di 720 Kbyte, ne potremo utilizzare solo 360.

Ma la prova che più di tutte le altre ha dato soddisfazione è stata quella effettuata con il GW Basic; il funzionamento è stato perfetto al punto da consentire la stesura di un vero programma di Benchmark vali-

UN DRIVE UNIVERSALE

I lettori più smaliziati sanno che il "formato" dei dati utilizzato da Amiga è diverso da quello utilizzato dal sistema MS-Dos.

Sorge allora spontanea la questione su come sia possibile che i due formati coesistano nello stesso drive.

La risposta alquanto semplice e risiede nella tecnica costruttiva che caratterizza i drive dell'attuale generazione.

Si tratta, in pratica, di drive "passivi" (o stupidi) che possono essere pilotati via software adeguandosi alle specifiche esigenze di formato; è infatti nota la possibilit , da parte di Amiga, di leggere dischetti Atari, MSX, Yamaha ecc.

I nuovi drive non possiedono al loro interno un microprocessore nè un sistema operativo. Al corretto funzionamento della periferica provvede quindi il computer che, mediante opportuni comandi, sposta la testina in una zona specifica del disco, legge o scrive dati e provvede, insomma, alla corretta gestione delle informazioni.

E' quindi sufficiente scrivere software adeguato per essere in grado di leggere dischetti provenienti da qualsiasi computer. A patto, lo ripetiamo, di sapere in che modo i vari dati sono stati memorizzati sulla superficie magnetica.

do sia per Amiga sia per Amstrad PPC 640 D.

Il programma in questione effettua tre tipi di prova: una prova di scrittura a video, una di scrittura a file ed una di lettura da file; il programma è ampiamente commentato e non necessita di nessun particolare chiarimento; uno sguardo ai risultati è invece d'obbligo.

La differenza più rilevante si riscontra a livello video, a causa della differente gestione fra le due macchine.

A parte la lentezza dovuta all'interpretazione da parte del Turbo Transformer, un altro fattore ha influito sulla velocità d'esecuzione: l'Amiga lavora SEMPRE in grafica; in pratica non conosce il "modo Testo" tipico dei PC.

Per quanto riguarda le operazioni di I/O da file il tempo impiegato da Amiga non è eccessivo se si considera l'emulazione software.

Un'ultima considerazione riguarda il colloquio tra Amiga e stampante; in molti casi il Sistema Operativo risponde con un "Device Fault" che segnala qualche problema di trasferimento dati.

CONCLUSIONE

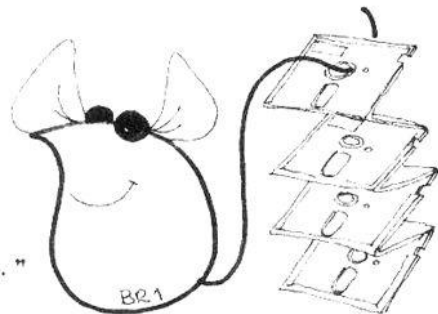
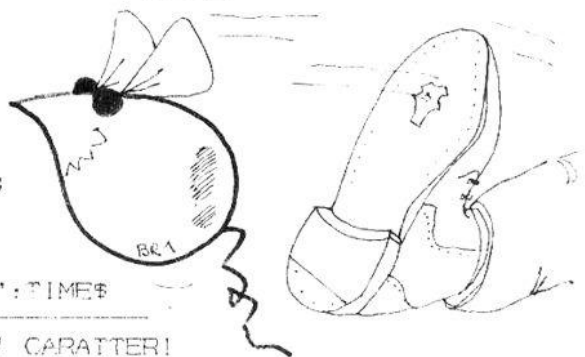
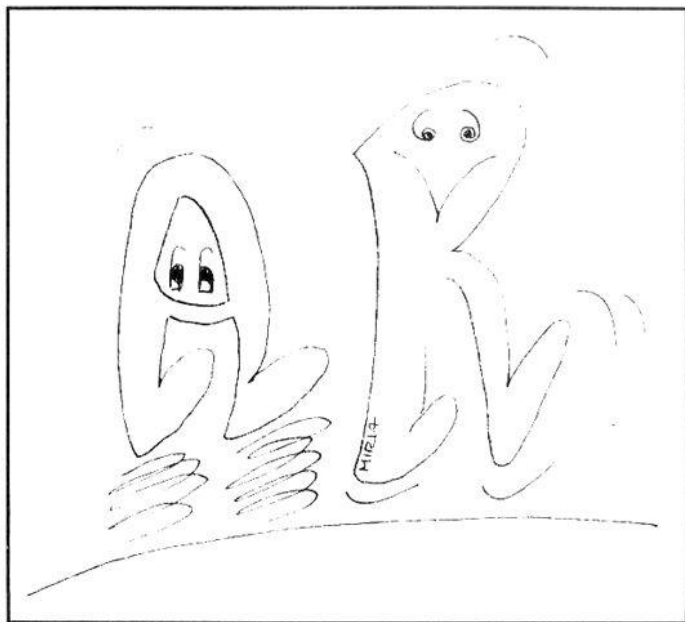
Tirando le somme, sia pur con qualche riserva, il giudizio non può che essere positivo; le possibilità dell'emulatore vanno ben oltre la semplice directory o il banale CLS; la possibilità di trasferire file in formato MS-Dos da Amiga a PC è una possibilità assai interessante che apre nuovi orizzonti per gli "smanettoni".

In conclusione, se avete Turbo Transformer tenetevelo ben stretto. Se non lo possedete procuratevelo: il fantastico mondo MS-Dos è sempre più vicino.

```
PC Tools Deluxe R4.22
------------------------------System Information Service-----


                   Operating system - DOS 3.30
     Number of logical disk drives - 5
          Logical drive letter range - A thru E
                        Serial Ports - 2
                      Parallel Ports - 1
                            CPU Type - NEC V20
     Relative speed (orig PC=100%) -  10%
          Math co-processor present - No
User programs are loaded at HEX paragraph - 0F25
 Memory used by DOS and resident programs - 62032 bytes.
      Memory available for user programs - 447920 bytes
          Total memory reported by DOS - 498K
PC Tools has found the total memory to be - 512K
      Monochrome Display Adapter present
```

Il "responso" di PC Tools

```
100 '*-----------------------------------------*
110 ' BENCHMARK PER AMIGA CON TRANSFORMER 1.2
120 ' IN COMPARAZIONE CON AMSTRAD PPC 640
130 ' AUTORE: MICHELE MAGGI
140 ' DATA: 09.03.1989
150 '*-----------------------------------------*
160 ' STAMPA 10 DIFFERENTI SCHERMATE
170 ' IN MODO TESTO
180 '*-------------------
190 KEY OFF
200 TIME$="000000"
210 CLS
220 FOR I=1 TO 10
230     FOR J=1 TO 2000
240         PRINT CHR$(64+I);
250     NEXT J
260     CLS
270 NEXT I
280 PRINT "TEMPO IMPIEGATO: ";TIME$
290 '*-----------------------------------------
300 ' SCRIVE 5000 STRINGHE DI CARATTERI
310 ' SU DISCHETTO DA 3.5 POLLICI
320 '*-----------------------------------------*
330 PRINT "TEST DI SCRITTURA SU DISCO"
340 TIME$="000000"
350 OPEN "AMIGA.TST" FOR OUTPUT AS £1
360 FOR I=1 TO 5000
370     PRINT £1,"COMMODORE COMPUTER CLUB"
380 NEXT I
390 PRINT £1,"SYSTEMS EDITORIALE"
400 CLOSE £1
410 PRINT "TEST DI SCRITTURA TERMINATO...."
420 PRINT "TEMPO IMPIEGATO: ";TIME$
430 '*-----------------------------------------*
440 ' LEGGE 5000 STRINGHE DI CARATTERI
450 ' SU DISCHETTO DA 3.5 POLLICI
460 '*-----------------------------------------
470 PRINT "TEST DI LETTURA DA DISCO"
480 TIME$="000000"
490 OPEN "AMIGA.TST" FOR INPUT AS £1
500 FOR I=1 TO 5000
510     INPUT £1, A$
520 NEXT I
530 PRINT "TEST DI LETTURA TERMINATO..."
540 PRINT "TEMPO IMPIEGATO: ";TIME$
```

Il listato del benchmark, scritto in GW Basic

CARATTERI... INFERMI!

Un insolito brulichio tenderà ad animare lo schermo del vostro C/16 - Plus/4

di **Roberto Ferro**

Il sogno di ogni sedicista o plusquattrista è sempre stato quello di poter disporre degli sprite, quei fantastici oggetti grafici che hanno fatto grande, e fanno tutt'ora, la grafica di computer come il C/64 e l'Amiga.

La carenza si è sempre fatta sentire in maniera più o meno pesante, sopratutto nelle varie realizzazioni di videogame che, nella conversione da C/64 a C/16, perdevano gran parte del loro fascino dovuto, appunto, agli sprite. Una tale situazione, però, è stata mal sopportata dai programmatori di C/16, i quali si sono ingegnati nei modi più diversi per cercare di sopperire a questa lacuna.

Sul numero 54 di Commodore Computer Club, ad esempio, abbiamo già studiato una particolare tecnica di sfruttamento del raster register per ottenere animazioni con movimento pixel per pixel.

Questo mese ci cimenteremo, invece, in un'impresa un po' diversa (ed invero anche un po' meno complicata), cercando però di ottenere, come fine ultimo, quello di realizzare con il C/16 scrolling di tipo cosiddetto "fine".

Ci serviremo della tecnica di ridefinizione dei caratteri, dell'interrupt e di qualche istruzione L.M. forse non troppo nota.

SCROLLING IN ROTAZIONE

Quando decidiamo di realizzare uno scrolling, la prima cosa da stabilire è il tipo di spostamento: bisogna cioè decidere come si comporterà la porzione di immagine che, a causa dello scrolling, non sarà più visibile.

Le possibilità che abbiamo sono due: o realizzare un effetto "rotazione" facendo riapparire da un lato ciò che scompare dall'altro, oppure dimenticarci della parte fuoriuscita e immettere nel nostro quadro una nuova porzione di immagine, realizzando in questo modo uno scrolling continuo.

Come è abbatanza facile intuire il primo metodo è di più facile realizzazione (ed è lo stesso che è stato utilizzato in ScrollPix sul numero di Giugno '88 di C.C.C.), e può presentare vantaggi in particolari situazioni.

Poniamo il caso di avere due monitor affiancati e di trasmettere, in ognuno di essi, la stessa immagine che si sposta in senso orizzontale. Se lo spostamento di queste immagini è del tipo "in rotazione" ed i due monitor sono strettamente affiancati, avremo l'impressione che l'immagine passi da uno schermo ad un altro, come se fossero un tutt'uno. Vedremmo, in pratica, sempre un'immagine completa, magari divisa tra l'uno e l'altro schermo. Questa tecnica viene spesso sfruttata nei mega screen plurivideo che compongono la scenografia di alcune trasmissioni televisive.

Il discorso vale anche per più di due monitor affiancati, in quanto maggiore sarà il numero dei monitor e maggiori "trasmigrazioni" effettuerà l'immagine da schermo a schemo.

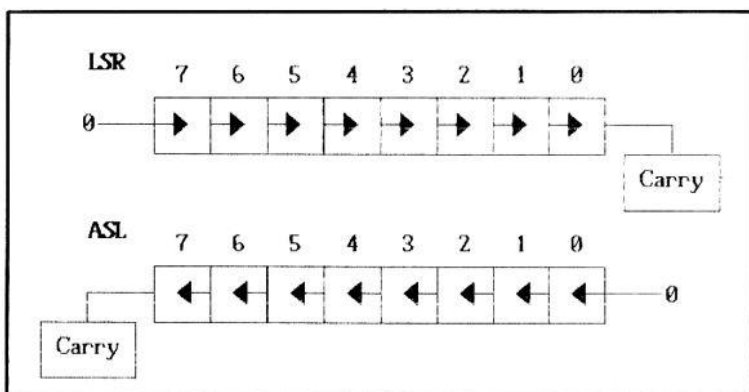
Ma che c'entrano gli schermi con il nostro C/16 o Plus/4 ? Vediamo.

Se proviamo ad immaginare la matrice pixel di un carattere (8 X 8) come un minuscolo monitor e l'insieme dei mille caratteri della memoria video come tanti schermi affiancati, ecco che si prospetta la possibilità di realizzare un'ottima "illusione" ottica di sicuro effetto per realzzare sfondi e paesaggi per i nostri programmi.

LA RIDEFINIZIONE E L'INTERRUPT

Il problema che però è necessario risolvere è quello di realizzare un'immagine in scrolling all'interno di una cella video. Per rimuovere tale ostacolo abbiamo due strumenti: la ridefinizione dei caratteri e l'interrupt (l'Onnipresente!).

Per ridefinire i caratteri, infatti, siamo costretti a trasferirne la mappa di riferimento da una zona di memoria ROM alla RAM, in modo che questa possa essere modificata a nostro piacimento. Il fatto, però, che la mappa dei caratteri venga a trovarsi in RAM consente ai caratteri di venir ridefiniti non una, ma quante volte vogliamo. Questo significa che possiamo realizzare un carattere in movimento limitandoci a ridefinirlo in continuazione.

Come viene brevemete spiegato nel riquadro, un carattere è costituito da otto

bytes consecutivi, ogni bit dei quali corrisponde ad un pixel della matrice del carattere. Se prendiamo singolarmente ogni byte e ne spostiamo tutti i bit verso destra con l'istruzione L.M. LSR (oppure ASL per lo spostamento verso sinistra) avremo il bit numero 7 (il più sigificativo) posto a zero, mentre il bit 0 "cade" nel carry, come mostra la figura 1. Effettuato lo spostamento, verifichiamo il contenuto del carry; se questo è settato a 1 allora settiamo a 1 anche il bit 7 del byte in quesione, realizzando la perfetta rotazione di tutti i bit.

Applicando lo stesso meccanismo agli otto bytes della matrice del carattere si avrà l'effetto di "rotazione" su tutto il carattere.

Questo è quanto accade per lo spostamento orizzontale. Per quanto concerne, invce, lo spostamento verticale è necessario arrangiarsi diversamente: dobbiamo provvdere noi stessi a ricopiare ogni byte su quello successivo ed a riposizionare l'ultimo al di sopra del primo, ottenendo la desiderata rotazione.

Per fare in modo che gli spostamenti siano continui, ci serviamo della frequenza dell'interrupt. La routine proposta in queste pagine lo devia in maniera che possa essere eseguito lo scrolling prima di riprendere il normale corso. Per dosare la velocità di scrolling sono stati usati contatatori in pagina zero (e precisamente le locazioni $D0 e $D1) che solo dopo aver raggiunto un certo valore fanno scattare il meccanismo di scrolling (si tratta in pratica di un banale sistema di rallentamento).

SHIFT CHARACTER

Quanto è stato appena descritto è, in breve, la spiegazione di cosa fa la routine "Shift Character". Il programma Basic (che contiene in forma di DATA la routine) provvede solo ad abbassare il top di memoria, trascrivere la routine, verificare che tutto sia andato per il meglio e, in caso positivo, stampare l'intestazione con le SYS necessarie al funzionamento. A questo punto, però, attivando la routine con le SYS fornite non potremo vedere alcun effetto perchè è necessaria la procedura di ridefinizione dei caratteri non inclusa in "Shift Chracter".

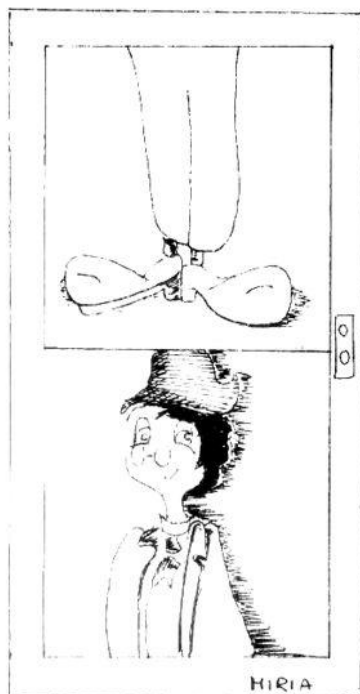
Per questo proposito è stata realizzata la routine "Demo Shift Character" che provvede a ridefinire i caratteri (o megio a ricopiare semplicemente la mappa ROM dei caratteri sulla RAM) e a realizzare uno "sfondo" in movimento sullo schermo del C/16.

Poichè hanno una pura funzione dimostrativa, le routine non provvedono a realizzare gli scrolling in tutte le diverse direzioni possibili, ma solo da sinistra a destra, dall'alto in basso e (combinando questi due spostamenti) in diagonale.

Sia i listati che i disassemblati sono ampiamente commentati: non dovrebbero esserci problemi a comprendere il funzionamento che, comunque, si basa sulla tecnica descritta.

E' bene ricordare che la tecnica descritta è un po'... delicata. E' quindi necessario evitare di commettere syntax error (o di altro tipo) che possono mandare in tilt la macchina. Il demo agisce sul carattere di chiocciolina che, tutte le volte che compare sullo schermo, subisce l'animazione descritta.

In caso di pasticci la tecnica è la solita: premere il tasto Reset mentre il tasto Run/Stop è tenuto abbassato. Digitare poi "X" per uscire dall'ambiente Monitor.

Un'ulteriore dimostrazione delle possibilità offerte da questo tipo particolare di scrolling la potete osservre sui muri dei recinti del gioco "Fury", per C/16 e Plus/4, pubblicato sul numero 17 di Software Club. In questo caso potrete notare come la linea diagonale, pur muovendosi solo in senso orizzontale, dia l'illusione di spostarsi anche verso il basso.

LA RIDEFINIZIONE DEI CARATTERI

La tecnica che va sotto il nome di ridefinizione dei caratteri è uno degli stratagemmi più usati in assoluto nella programmazione di videogame. Consiste nell'alterare le diverse lettere che appaiono sullo schermo, sostituendole con disegni, o parti di disegno, in modo da crearci incantevoli scenari, micidiali astronavi, piccoli mostriciattoli o anche solo un font di caratteri personalizzato. Ciò è possibile grazie al fatto che ogni carattere è, in fondo, un piccolo disegno creato utilizzando una matrice di 8 pixel per 8.

Tutti i dati relativi ai caratteri che vediamo normalmente sono scritti in ROM, in una zona specifica lunga 2 K che il computer considera come la "mappa dei caratteri", suddivisa in blocchi di 8 bytes, ognuno dei quali contiene i dati necessari a definire un carattere.

Grazie ad uno dei registri del TED Chip, possiamo "raggirare" il Sistema Operativo e fargli credere che tale mappa si trovi in RAM. Grazie, poi, ad un altro registro del TED, possiamo anche indicare la zona che più ci aggrada.

I registri in questione sono il 18 ed il 19 ($FF12 - 65298 e $FF13 - 65299). In particolare il bit 2 del registro 18 ($FF12) effettua la selezione RAM / ROM (normalmente è posto a 1) mentre i bit 7-2 del registro 19 ($FF13) decretano l'inizio del blocco di 2K destinato a contenere la mappa.

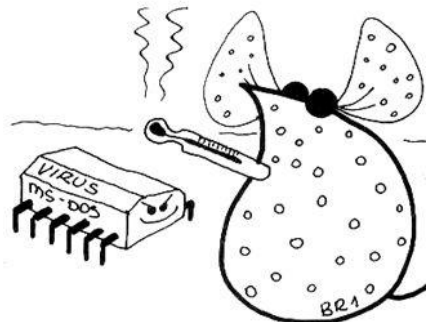
Come è già stato detto in altra parte dell'articolo, i bytes necessari a definire un carattere sono 8 e devono essere consecutivi; il primo di questi rappresenterà la prima fila di 8 pixel mentre gli altri, fino all'ultimo, rappresentano, via via, le otto linee di otto pixel ognuna. Assegnando ad ogni byte un determinato valore provocheremo l'accensione dei pixel relativi ai bit settati, e quindi potremo visualizzare il nostro disegno.

Notate come i bit del registro 19 relativi all'indirizzo di partenza della mappa caratteri siano solo 6, e pertanto possono rappresentare solo 64 valori diversi, uno per ogni K di memoria indirizzata dal 7510. Questo significa che non è possibile far cominciare la mappa dei caratteri da un byte qualsiasi, ma solo in corrispondenza dell'inizio di un nuovo Kappa di memoria.

Un'ultima osservazione va fatta circa il trasferimento della mappa caratteri dalla ROM alla RAM. Spesso succede che occorre ridefinire solo alcuni caratteri ed aver disponibili gli altri (come nel caso del Demo Shift Character che, addirittura, non ne ridefinisce neanche uno). In questo caso è conveniente ricopiare la mappa. C'è però un piccolo inconveniente in quanto la zona di memoria $D000 - $DFFF non può essere letta da Basic e si rende necessaria una breve routine L.M. (come quella facente parte di Demo Shift Character) che assolverà a questa funzione.

```
100 REM ***************************
110 REM       PROGRAMMA DEMO PER
120 REM        SHIFT CHARACTER
130 REM ***************************
140 :
150 FOR I=12288 TO 12319
160 READ A$:POKEI,DEC(A$)
170 CK=CK+DEC(A$)
180 NEXT I
190 IF CK<>3379 THEN 430
200 :
210 REM COPIA I CARATTERI IN RAM
220 SYS 12288
230 REM ATTIVA IL SET DI CARATTERI RAM
240 POKE 65298,192:POKE 65299,57
250 :
260 FORI=3072 TO 4071:POKEI,0:NEXT
270 CHAR 1,11,14," (D)ESTRA - (B)ASSO "
280 CHAR 1,11,15,"      (O)BLIQUO      "
290 GETKEYA$
300 IF A$="D" THEN SYS 14200
310 IF A$="B" THEN SYS 14214
320 IF A$="O" THEN SYS 14228
330 GOTO 290
340 :
350 REM ROUTINE PER COPIARE LA ROM
360 REM DEI CARATTERI SULLA RAM
370 REM ALL'INDIRIZZO $3800
380 :
390 DATA A2,00,BD,00,D0,9D,00,38
400 DATA BD,00,D1,9D,00,39,BD,00
410 DATA D2,9D,00,3A,BD,00,D3,9D
420 DATA 00,3B,E8,D0,E5,60,00,00
430 PRINT"ERRORE NEI DATA !!!"
440 END
```

```
10 REM ***************************
20 REM        SHIFT CHARACTER
30 REM   SOLO PER C/16 & PLUS/4
40 REM ***************************
50 :
60 REM        BY ROBERTO FERRO
65 REM PER COMMODORE COMPUTER CLUB
70 :
80 POKE55,119:POKE56,55
100 FOR I=14200 TO 14331
110 READ A$:POKEI,DEC(A$)
120 CK=CK+DEC(A$)
130 NEXT I
140 IF CK<>14042 THEN 200
145 SCNCLR
150 PRINTCHR$(17)" > SHIFT CHARACTER:"
155 PRINTCHR$(17)" "CHR$(18)"SYS 14200"
156 PRINT"  MOVIMENTO VERSO DESTRA"
157 PRINTCHR$(17)" "CHR$(18)"SYS 14214"
158 PRINT"  MOVIMENTO VERSO IL BASSO"
159 PRINTCHR$(17)" "CHR$(18)"SYS 14228"
160 PRINT"  MOVIMENTO DIAGONALE"
170 END
200 SCNCLR
210 PRINT"ERRORE NEI DATA !!!":END
1000 :
1001 REM ROUTINES DI DEVIAZIONE
1002 REM DELL'INTERRUPT
1003 :
1010 DATA 78,A9,A7,8D,14,03,A9,37
1011 DATA 8D,15,03,4C,9F,37
1012 DATA 78,A9,AD,8D,14,03,A9,37
1013 DATA 8D,15,03,4C,9F,37
1014 DATA 78,A9,B3,8D,14,03,A9,37
1015 DATA 8D,15,03
1016 :
1035 REM ROUTINE PER L'AZZERAMENTO
1036 REM DEI CONTATORI IN PAGINA 0
1037 :
1038 DATA A9,00,85,D1,85,D2,58,60
1039 :
1110 REM ROUTINES IN INTERRUPT PER IL
1120 REM CONTROLLO DEL MOVIMENTO
1130 :
1140 DATA 20,BC,37,4C,0E,CE
1150 DATA 20,DC,37,4C,0E,CE
1160 DATA 20,BC,37,20,DC,37,4C,0E,CE
1170 :
1180 REM SUBROUTINE PER IL MOVIMENTO
1190 REM VERSO DESTRA
1200 :
1210 DATA E6,D0,A5,D0,C9,06,D0,17
1220 DATA A9,00,85,D0,18,A2,00,BD
1230 DATA 00,38,4A,90,02,09,80,9D
1240 DATA 00,38,E8,E0,08,D0,F0,60
1250 :
1260 REM SUBROUTINE PER IL MOVIMENTO
1270 REM VERSO IL BASSO
1280 :
1290 DATA E6,D1,A5,D1,C9,06,D0,17
1300 DATA A9,00,85,D1,AD,07,38,48
1310 DATA A2,06,BD,00,38,9D,01,38
1320 DATA CA,10,F7,68,8D,00,38,60
1330 END
```

```
*-----------------------------------*
*ROUTINE  SHIFT CHARACTER  per C/16 *
*by ROBERTO FERRO           & Plus/4 *
*-----------------------------------*

3778  78        sei        ;devia
3779  a9 a7     lda #$a7   ;l'interrupt
377b  8d 14 03  sta $0314;per puntare
377e  a9 37     lda #$37   ;alla routine
3780  8d 15 03  sta $0315;scroll right
3783  4c 9f 37  jmp $379f
3786  78        sei
3787  a9 ad     lda #$ad
3789  8d 14 03  sta $0314
378c  a9 37     lda #$37
378e  8d 15 03  sta $0315;scroll left
3791  4c 9f 37  jmp $379f
3794  78        sei
3795  a9 b3     lda #$b3
3797  8d 14 03  sta $0314
379a  a9 37     lda #$37   ;scroll in
379c  8d 15 03  sta $0315;diagonale
379f  a9 00     lda #$00   ;azzera
37a1  85 d1     sta $d1    ;registri in
37a3  85 d2     sta $d2    ;pagina zero
37a5  58        cli
37a6  60        rts
37a7  20 bc 37  jsr $37bc
37aa  4c 0e ce  jmp $ce0e
37ad  20 dc 37  jsr $37dc
37b0  4c 0e ce  jmp $ce0e
37b3  20 bc 37  jsr $37bc
37b6  20 dc 37  jsr $37dc
37b9  4c 0e ce  jmp $ce0e
37bc  e6 d0     inc $d0    ;inizio routine
37be  a5 d0     lda $d0    ;scrolling a
37c0  c9 06     cmp #$06   ;destra
37c2  d0 17     bne $37db
37c4  a9 00     lda #$00
37c6  85 d0     sta $d0
37c8  18        clc
37c9  a2 00     ldx #$00     ;sposta di
37cb  bd 00 38  lda $3800,x;un pixel
37ce  6a        lsr          ;e controlla
37cf  90 02     bcc $37d3    ;il carry
37d1  09 80     ora #$80     ;pone a 1 il
37d3  9d 00 38  sta $3800,x;bit sette
37d6  e8        inx
37d7  e0 08     cpx #$08
37d9  d0 f0     bne $37cb;ciclo
37db  60        rts        ;ritorno
37dc  e6 d1     inc $d1    ;inizio routine
37de  a5 d1     lda $d1    ;scrolling in
37e0  c9 06     cmp #$05   ;basso
37e2  d0 17     bne $37fb
37e4  a9 00     lda #$00
37e6  85 d1     sta $d1
37e8  ad 07 38  lda $3807;pone il byte
37eb  48        pha        ;piu' alto
37ec  a2 06     ldx #$06   ;nello stack
37ee  bd 00 38  lda $3800,x;trasferisce
37f1  9d 01 38  sta $3801,x;tutti i byte
37f4  ca        dex        ;una posizione
37f5  10 f7     bpl $37ee;piu' in basso
37f7  68        pla        ;recupera il
37f8  8d 00 38  sta $3800;byte dallo
37fb  60        rts        ;stack
```

```
*---------------------------------------*
* ROUTINE DEMO SHIFT:                   *
*                   TRANSFER CHARACTER  *
*---------------------------------------*

3000  a2 00     ldx #$00    ;trasferisce
3002  bd 00 d0  lda $d000,x;i caratteri
3005  9d 00 38  sta $3800,x;dalla
3008  bd 00 d1  lda $d100,x;ROM ($d000)
300b  9d 00 39  sta $3900,x;alla
300e  bd 00 d2  lda $d200,x;RAM ($3800)
3011  9d 00 3a  sta $3a00,x
3014  bd 00 d3  lda $d300,x
3017  9d 00 3b  sta $3b00,x
301a  e8        inx
301b  d0 e5     bne $3002
301d  60        rts
```

CAMPUS

LABORATORIO SOFTWARE DI COMMODORE COMPUTER CLUB

ESORDIENTI

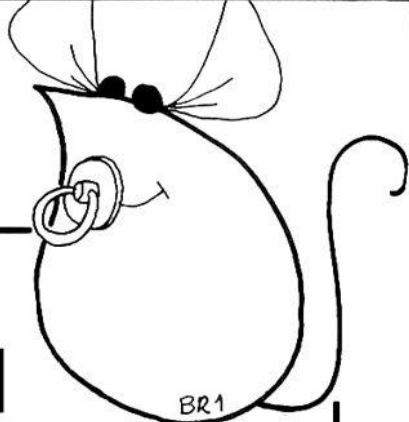
- C/64, la tua Settimana Enigmistica su video
- Un orologio a spasso sul video

ESPERTI

- Abbattete le barriere

CAMPIONI

- Come gestire 64 sprite
- Una corsa in videomobile

PRINCIPIANTI, I DIECI COMANDAMENTI

1 Leggi attentamente il libretto di istruzioni del tuo computer e delle periferiche che ti sei procurato (registratore, drive, stampante, monitor, eccetera)

2 Non limitarti a leggere i listati di esempio ivi riportati, anche se sembrano banali: digitali sul computer e falli girare (digitando Run e premendo il tasto Return). Se non ti sono chiari, al contrario, la loro digitazione (e successiva esecuzione) ti chiarirà le idee.

3 Nel digitare i listati, ricordati di premere SEMPRE il tasto Return quando giungi alla fine del rigo, anche se tale operazione può sembrarti inutile.

4 Non confondere la vocale alfabetica "O" con il numero zero "0".

5 Digita sempre per esteso il comando PRINT e non abbreviarlo mai con il punto di domanda (?).

6 Inizia a digitare i listati più semplici e brevi: quelli più lunghi potrai digitarli quando avrai acquisito una maggior dimestichezza con il computer.

7 Dopo aver digitato un QUALSIASI programma, registralo subito, seguendo le istruzioni riportate sul manuale, PRIMA di dare Run.

8 Dopo aver fatto partire un programma, in caso, ad esempio, di segnalazione di errore in linea 350, digita soltanto...
List 350
...e accertati che la linea che appare sul video sia RIGOROSAMENTE identica a quella stampata sulla rivista.

9 Se, digitando un listato, ti accorgi che vi sono istruzioni di tipo Poke e Sys, raddoppia la prudenza nella digitazione e nelle procedure di registrazione PRIMA di dare Run.

10 Accertati che il programma che ti accingi a digitare sia REALMENTE valido per il tuo computer. Su ogni articolo della nostra rivista (da leggere SEMPRE con attenzione) è indicato il tipo di computer per il quale il listato stesso è idoneo.

C/64, LA TUA "SETTIMANA ENIGMISTICA" SU VIDEO

Poche routine ti permettono di trascorrere alcune ore liete in compagnia dei tuoi amici

di **Alessandro de Simone**

Esattamente un anno fa (N. 52) ci siamo occupati della codificazione dei messaggi segreti. I programmi pubblicati, grazie ad una tecnica oltremodo banale, hanno avuto molto successo in tutto il mondo tanto che numerosi agenti segreti (007 in testa) li usano correntemente, sicuri di essere al riparo da sguardi indiscreti.

Stavolta ricorriamo ancora alla tecnica della crittografia, ma solo per divertirci e per dimostrare ancora una volta che risulta piuttosto semplice trasferire su computer (o, come si dice in grego, "implementare") tecniche e procedimenti usati in ben altri campi.

Ci stiamo riferendo al famoso settimanale "La Settimana Enigmistica" e, in particolare, ai giochi della serie "Aneddoti Cifrati" in cui vengono riportati alcuni numeri che rappresentano, ciascuno, un carattere dell'alfabeto. Al giocatore spetta il compito, ovviamente, di rintracciare il codice e di risolvere il gioco.

Si guardino, ad esempio, le due tabelle in cui è riportato un esempio chiarificatore, tratto dal n. 2971 del popolare periodico.

Nella prima tabella è riportato lo schema del gioco, così come si trova sulla "Settimana Enigmistica". Si notino i vari codici (il primo è sempre il n. 1), i trattini (che rappresentano la separazione tra parole successive), ed alcuni termini ("attrice", nell'esempio) riportati "in chiaro" per facilitare lo svolgimento del gioco.

Nella seconda tabella è riportata la soluzione.

Ad ogni numero, quindi, corrisponde un carattere: non vi sono segni di punteggiatura (nè apostrofi). Ne consegue che il valore massimo assunto dal codice può essere rappresentato dal 26, ammesso (e non concesso) che la frase sia tanto lunga e complessa da tirare in ballo tutte le lettere dell'alfabeto.

Nel caso in cui alcuni caratteri si ripetono molto frequentemente, il valore massimo sarà modesto. E' ovvio che non vi possono esser "salti" di codice: se in un aneddoto cifrato è presente, da qualche parte, il n. 16 devono esser presenti tutti i precedenti, dall'1 al 15 compresi.

Ciò si verifica perchè la tecnica della codifica è tanto banale quanto efficace: si inizia attribuendo al primo carattere della frase il n. 1; al successivo carattere, se diverso dal precedente, si assegna il primo valore disponibile (cioè il N. 2) e così via fino all'ultimo carattere.

Tutti i n. 1, insomma, sono eguali tra loro (nell'esempio riportato a 1 corrisponde "I", 2 coincide con "N" e così via). La correttezza della decodifica sarà confermata dall'apparizione di una frase non priva di senso.

Trasferire su computer un gioco enigmistico è relativamente semplice

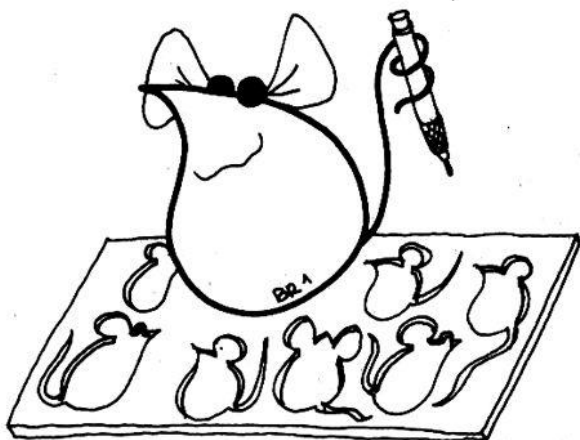
IL GIOCO NEL COMPUTER

La semplicità della tecnica di codifica consente di trasferire, sul vostro calcolatore, il procedimento descritto utilizzando una manciata di righe Basic.

LE AVVENTURE DI

GIOVEDINI

by Marco Mietta
Andrea Paolini

VOLA AL CINEMA !

Dal momento che, però, ci rivolgiamo soprattutto a coloro che vogliono imparare (più che a coloro che vogliono giocare) è opportuna una prima descrizione delle routine usate.

Il primo programma di queste pagine ("Assegnazione Codice") rappresenta la routine di base che verrà successivamente usata nel programma completo.

Abbiamo prima detto che il massimo numero di codici che è possibile creare è 26, tante quante sono le lettere dell'alfabeto (comprese X, Y, J, K, W). Ad ognuna di queste corrisponderà un numero che possiamo ritenere casuale perchè l'assegnazizone dipenderà dalla posizione dei caratteri costituenti le singole parole.

Nel primo listato, in riga 200, è contenuta la frase, da visualizzare in codice, che viene letta dal comando Read di riga 140. Poichè non è possibile conoscere a priori se è presente una qualche ripetizione di un carattere, non è possibile (o meglio, risulta troppo complicato) generare una matrice in cui siano contenuti solo i caratteri costituenti la frase da crittografare ed i corrispondenti codici.

E' molto più semplice, infatti, creare una matrice bidimensionale (A%), lunga 26 elementi, in cui allocare i codici ASCII degli altrettanti caratteri alfabetici. "Parallelamente" (vale a dire nella seconda colonna della stessa matrice) verranno trascritti i vari codici associati ai caratteri, altrimenti sarà presente il valore nullo.

Nella tabella, relativa alla frase "Leggete le riviste Systems", si può notare che al primo carattere ("L", codice ASCII 76) corrisponde, appunto, il codice 1; al secondo ("E" codice ASCII 69) il 2 e così via fino all'ultimo ("M", Ascii 77) che è il n. 10. L'intera frase, infatti, è formata da 10 caratteri diversi tra loro (vedi figura 3).

Facendo girare il semplice programma (ricordandosi, ovviamente, di cambiar frase in riga 200 prima di dare Run ogni volta), capirete meglio la tecnica descritta.

GIOCHIAMO

Passiamo ora alla descrizione del secondo listato, che costituisce il gioco vero e proprio.

Per facilitare la correttezza della digitazione, da parte del lettore, è prevista un'apposita opzione che, selezionando "Prova" o "Frase" (riga 140) permetterà di verificare che tutto funzioni a dovere. Nel primo caso, infatti, verrà testata la frase trascritta in riga 380.

Giocando in due, invece, è necessario che uno

```
100 REM PRIMA SUBROUTINE:
110 REM ASSEGNAZIONE CODICE A CIASCUN CARATTERE
120 REM DELLA FRASE DA CIFRARE
130 :
140 READ B$: L-LEN(B$)
150 DIM A%(26,2): FOR I-65 TO 90: A%(I-64,0)=I: NEXT
160 U-1: FOR I-1 TO L
170 Y$- MID$(B$,I,1): Y-ASC(Y$):IF Y-32 THEN 190
180 IF A%(Y-64,1)-0 THEN A%(Y-64,1)-U:U-U+1
190 NEXT: FORI-1 TO 26: PRINT A%(I,0);A%(I,1): NEXT
200 DATA LEGGETE LE RIVISTE SYSTEMS
```

Dopo aver risolto il caso degli "emulatori" del mese scorso, Primo Giovedini ha ripreso la normale attività di volo: ora infatti sta svolgendo una missione di routine (anzi, di "subroutine") di poca importanza...

```
100 REM * * * * *  CRITTOGRAMMI  * * * * *
110 REM PROGRAMMA PER PRINCIPIANTI
120 REM VERSIONE C/64, C/128, C/16, PLUS/4
130 :
140 PRINTCHR$(147):INPUT"PROVA O FRASE (P/F)";X$
150 IFX$<>"P"THEN GOSUB 500:GOTO190
160 READ B$
170 :
180 REM CODIFICAZIONE DELLA FRASE
190 DIM A%(26,2)
200 FOR I=65 TO 90:A%(I-64,0)=I: NEXT
210 L=LEN(B$):DIM B$(L)
220 U=1:FOR I=1 TO L
230 Y$= MID$(B$,I,1): Y=ASC(Y$):IF Y=32 THEN 250
240 IF A%(Y-64,1)=0 THEN A%(Y-64,1)=U:U=U+1
250 NEXT: GOSUB410
260 :
270 REM RICHIESTA ASSEGNAZIONE CODICE-CARATTERE
280 PRINTCHR$(19);: INPUT"NUMERO";NU:INPUT"CARATTERE";NU$
290 :
300 REM VERIFICA ESISTENZA CODICE-CARATTERE
310 P=0:FORK=1TO L:IF B$(K)=NU$ THEN P=1
320 NEXT:IFP=1 THEN GOSUB 550:GOTO280
330 :
340 REM DEFINITAVA ASSEGNAZIONE CODICE-CARATTERE
350 B$(NU)=NU$:GOSUB410:GOTO280
360 :
370 REM FRASE-DATA DI PROVA
380 DATA"LEGGETE COMMODORE COMPUTER CLUB"
390 :
400 REM VISUALIZZAZIONE SU DUE COLONNE
410 PRINTCHR$(147):PRINT:U=0:Z=0:FOR I=1 TO L
420 Y$= MID$(B$,I,1): Y=ASC(Y$)
430 IF Y=32 THEN PRINT TAB(U);"***": GOTO 470
440 PRINT TAB(U);RIGHT$(CHR$(32)+STR$(A%(Y-64,1)),2);CHR$(32);
450 PRINT CHR$(18);B$(A%(Y-64,1))
460 IF I>20 AND Z=0 THEN PRINTCHR$(19):PRINT:U=10:Z=1
470 NEXT:RETURN
480 :
490 REM DIGITAZIONE FRASE DA CIFRARE
500 PRINTCHR$(147)"ALLONTANA GLI OCCHI INDISCRETI E DIGITA"
510 PRINT"LA FRASE DA INDOVINARE (MAX 42 CARATTERI)"
520 PRINT:PRINT:INPUT B$:IF LEN(B$)<43 THEN RETURN
530 GOTO500
540 REM SEGNALE DI CODICE GIA' ESISTENTE
550 PRINT CHR$(147);CHR$(18);NU$;CHR$(146);" ESISTE GIA'"
560 PRINT:PRINT"PREMI UN TASTO PER CONTINUARE
570 :
580 REM ATTESA PRESSIONE TASTO
590 GET NU$:IF NU$="" THEN 590
600 GOSUB 410:RETURN
```

```
1, 2 - 3, 2 - 4, 5, 6, 7, 8, 8, 7 -
3, 2 attrice 2, 7, 2 - 9, 1, 7
10, 1, 7, 11, 5, 2, 12 - 13, 1, 4, 4, 12
```

Figura 1: Parte di un aneddoto cifrato

dei due giocatori si allontani (righe 490 - 530) per consentire all'altro di digitare una frase (lunga al massimo 42 caratteri, spazi compresi) al cui interno siano assenti i tipici caratteri che non possono essere accettati dall'istruzione Input, come il doppio punto (:) e la virgola (,).

Non appena si preme un tasto comparirà la frase, in codice, lungo due colonne affiancate; per evidenziare eventuali spazi tra una parola e l'altra verrà trascritta, automaticamente, una terna di asterischi (riga 430).

Sulla sommità del video compare quindi la richiesta del numero cui desiderate attribuire un carattere e, subito dopo, la richiesta del carattere da associare al numero stesso.

Premuto il tasto Return lo schermo viene cancellato e (subroutine 400 - 470) in corrispondenza del numero precedentemente indicato, verrà posto il carattere scelto.

Si ripeterà, quindi, la stessa operazione per tutti i codici presenti, fino all'individuazione della frase cifrata.

Si noti che un opportuno filtro (subroutine 540 - 560) è stato posto per impedire che uno stesso carattere sia attribuito a due codici diversi.

Volendo, quindi, modificare un codice già "occupato", è sufficiente assegnare, a tale codice, un qualsiasi carattere non alfabetico (quelli numerici vanno benissimo) ed in seguito ripetere l'operazione assegnando il carattere definitivo.

Esempio:

Numero? 12
Carattere? C

Numero? 9
Carattere? C

C Esiste già

Numero? 12
Carattere? 7

Numero? 9
Carattere? C

A questo punto, ovviamente, dovrete assegnare a 12 un nuovo carattere dal momento che, per favorire lo "scambio", al 12 verrà ora affiancato il carattere "7", privo di senso.

NEI DETTAGLI

Chi desidera meglio apprendere la tecnica di programmazione adoperata potrà leggere le note che seguono; gli altri possono iniziare subito a digitare il listato (e a divertirsi).

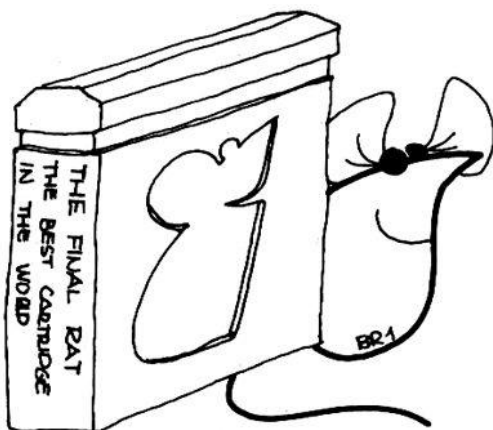
Nonostante le variabili in gioco siano poche, è stata scelta, per memorizzare i 26 caratteri alfabetici ed i corrispondenti codici, una matrice bidimensionale di tipo intero anzichè "normale" (cioè in virgola mobile).

La memoria occupata da ciascun elemento di una matrice intera richiede, infatti, solo due byte, anzichè cinque. Imparare a risparmiar byte può sempre essere utile quando realmente si avrà bisogno di esser tirchi...

Nel vettore B$ (riga 210) vengono allocati, l'uno di seguito all'altro, tutti i caratteri che, appartenenti alla frase B$, compaiono almeno una volta. Il vettore B$ sarà quindi utile per verificare eventuali involontari tentativi, da parte dell'utente, di assegnare uno stesso carattere a due codici diversi (righe 300 - 320 e subroutine 540, già esaminata precedentemente). Si noti la funzione del deviatore "P" che, posto inizialmente a zero, viene settato ad "1" nel caso in cui incontri un carattere già utilizzato.

Per ciò che riguarda la visualizzazione, notiamo

```
I  N  -  U  N  -  S  A  L  O  T  T  O
1  2  -  3  2  -  4  5  6  7  8 -8  7
---------------------------------------

U  N  ATTRICE  N  O  N  -
3  2  attrice  2  7  2  -

---------------------------------------

P  I  U  -   G  I  O   V  A  N  E
9  1  7  -  10  1  7  11  5  2  12

---------------------------------------

 D  I  S  S  E
13  1  4  4  12
```

Figura 2: Decodifica della frase

che questa risulta più agevole in verticale anzichè in orizzontale. La lettura finale, da parte del giocatore, ne risente un po' ma ci si abitua in fretta.

ASCII	Cod.				
A	65	0	N	78	0
B	66	0	O	79	0
C	67	0	P	80	0
D	68	0	Q	81	0
E	69	2	R	82	5
F	70	0	S	83	8
G	71	3	T	84	4
H	72	0	U	85	0
I	73	6	V	86	7
J	74	0	W	87	0
K	75	0	X	88	0
L	76	1	Y	89	9
M	77	10	Z	90	0

Figura 3: Codici della frase: "Leggete le riviste Systems"

Anche in questo caso la variabile "Z" assume la funzione di deviatore: per evitare scrolling indesiderati del video, giunti alla 25 riga la visualizzazione riprende dalla terza riga in alto, a distanza di ancuni byte dal bordo sinistro. Un attento studio delle righe da 400 a 470 fornirà delucidazioni sulla tecnica usata.

MIGLIORIE

A parte il doveroso controllo sulla esistenza di un codice già assegnato, non vi sono altri "filtri" che impediscano interruzioni dell'elaborazione o altre anomalie.

Suggeriamo di inserire, ad esempio:

- Input controllato che escluda (riga 280 e 520) la digitazione dei caratteri "vietati".
- Esclusione di valori di NU inadeguati (variabile NU, riga 280).
- Conteggio dei tentativi effettuati, con esclusione delle "sostituzioni" di caratteri già codificati.
- Interruzione del gioco, con assegnazione del punteggio corrispondente, nel caso si riesca ad individuare la frase crittografata.
- Introduzione di un cronometro e di un limite massimo di tentativi.

Altre migliori, come solito, potranno essere apportate dai lettori più in gamba.

Un qualsiasi programma è costituito da gruppo di subroutine

Pronto, base.
C'E' DA SPOSTARE UN AEREO! VENGA FUORI QUALCUNO, CHE DA SOLO NON CE LA FACCIO!

Subito parte un veicolo di servizio ...
Non preoccuparti, Primo: sistemeremo tutto noi!

UN OROLOGIO A SPASSO SUL VIDEO

Una breve utility visualizza, sullo schermo hi-res del C/128, un doppio orologio digitale / analogico

di **Leo Buffa**

L'orologio dimensionabile per CBM 128 ha questo nome perchè può variare le sue dimensioni ed inoltre può essere spostato dove si vuole nello schermo in alta risoluzione del C/128.

Il programma sfrutta, per la visualizzazione delle lancette, la routine dell'orologio pubblicato su C.C.C. N. 55.

All'inizio viene richiesta l'ora corrente; subito dopo si passa in alta risoluzione ed il quadrante viene visualizzato in alto a sinistra.

Questo si può spostare a piacere sullo schermo facendo uso dei soliti tasti di movimento del cursore.

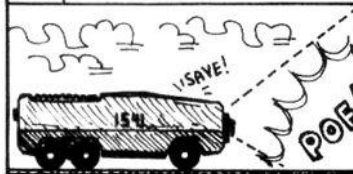
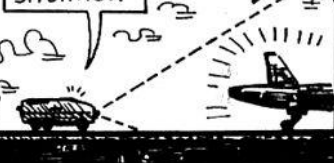
Durante il movimento il cerchio che compone l'orologio non viene disegnato per ovvi motivi di velocità. Anche quando si vorranno modificare le dimensioni per mezzo dei tasti (+) e (-) l'orologio verrà individuato da due punti: il primo indica il centro, il secondo la grandezza.

La variabile K assume valore 1 ogni volta che si effettua una operazione (di movimento, di ingrandimento o di riduzione).

Quando si rilasciano i tasti, si ritorna alla routine di GET grazie alla quale si controlla il valore di K.

Se K=1 viene ridisegnato il quadrante; in ogni altro caso si procede con la routine di movimento delle lancette.

Di proposito sono state evitate le routine di controllo di dimensioni e posizione dell'orologio a causa del notevole rallentamento che la procedura avrebbe comportato. Nulla vieta, però, di studiare metodi di controllo efficaci.

```
10 SCNCLR
20 PRINT"  ┐┌ ┬┐      ┐┌┐┌┐
30 PRINT"  HEX||ATA    |┘H
40 PRINT"  ┘└ └┘      ┴└┘└┘
50 PRINT"* * * * * * * * * *"
60 INPUT"▪▪TIME";TI$
70 X=25:Y=35:R=15
80 GRAPHIC1,1
85 CIRCLE 1 ,100 ,100,15,40
90 GOTO110
100 GRAPHIC1,1:DRAW1,X,Y:DRAW1,X+R,Y:RETURN
110 CIRCLE1,X,Y,R:K=0
```

```
120  :DO
130  :     GOTO150
140  :     GOSUB100
150  :    GETA$
160  :    IFA$=" "THEN270
170  :    IFA$="+"THENR=R+5:K=1:GOTO140
180  :    IFA$="-"THENR=R-5:K=1:GOTO140
190  :    IFA$="▌"THENX=X+5:K=1:GOTO140
200  :    IFA$="▐"THENX=X-5:K=1:GOTO140
210  :    IFA$="◙"THENY=Y+5:K=1:GOTO140
220  :    IFA$="]"THENY=Y-5:K=1:GOTO140
230  :    IFA$=""ANDK=1THENGOSUB250
240  :    GOTO 270
250  :    REM FINITE LE MODIFICHE RIDISEGNA IL QUADRANTE
255  :    CIRCLE1,X,Y,R:K=0:RETURN
260  :::REM:RINNOVO:POSIZIONE:LANCETTE:::
270  :    DO
280  :      S$=RIGHT$(TI$,2):S=VAL(S$)
290  :    LOOPUNTILS<>51
300  :     CHAR,01,01,O$+":"+M$+":"+S$
310  :     REM LANCETTA SECONDI
315  :     DRAW0,X,YTOR-3;S1*6:DRAW1,X,YTOR-3;S*6:S1=S
320  :     M$=MID$(TI$,3,2):M=VAL(M$)+S/60
330  :     REM LANCETTA MINUTI
335  :     DRAW0,X,YTOR-R/25;M1*6:DRAW1,X,YTOR-R/25;M*6:M1=M
340  :     O$=LEFT$(TI$,2):O=VAL(O$)+M/60
350  :     REM LANCETTA ORE
355  :     DRAW0,X,YTOR-R/5;O1*30:DRAW1,X,YTOR-R/5;O*30:O1=O
360  :LOOP
```

Tutta la fretta di Giovedini ora si spiega: resettato il suo F-18, egli si reca velocemente alla sala proiezioni della portaerei, dove stanno per dare la prima visione del film del CAPITANO BLITTER...

A SUON DI ASSEMBLY

Sfruttiamo l'accoppiata vincente Macro Assembler - C/64 per gestire al meglio le capacità sonore del nostro computer

di **Domenico Pavone**

Finalmente la musica si sposa con l'Assembly!

Se si effettuasse un sondaggio statistico tra gli utenti del C/64, per determinare quali caratteristiche del computer affascinano maggiormente, il suono sarebbe di certo una delle più gettonate, seconda, forse, solo alla grafica.

Chi non si è stupito, almeno una volta, ascoltando la perfezione di alcune colonne sonore (è proprio il caso di definirle così) che accompagnano i top game in commercio?

Un rovescio della medaglia, però, esiste, ed è ben noto a chi non si accontenta solo di utilizzare passivamente il software degli altri; gli strumenti offerti all'aspirante programmatore per gestire simili potenzialità della macchina, sono decisamente pochi, e addirittura inesistenti a livello di comandi Basic.

L'unica via relativamente semplice per tirar fuori qualche nota dal computer, passa infatti attraverso una lunga serie di Poke, con risultati, nella migliore delle ipotesi, limitati.

Qualcosa di più sofisticato lo si può ottenere con l'aiuto dei molti programmi "dedicati" presenti in commercio (tipo Music Studio), o prelevando il sonoro da software non protetto (avete presente Software Club ed i dischetti della serie "Directory"?...), ma se poi si vuole inserire il brano in un proprio programma, magari in modo che lo si ascolti mentre lo stesso fa tutt'altro, si giunge sempre allo stesso inevitabile punto: l'obbligo di usare il linguaggio macchina.

In che modo, lo vedremo assieme tra breve, occupandoci degli aspetti pratici legati alla gestione del circuito integrato 6581, ovvero della sua attivazione e messa in relazione con l'interrupt di sistema.

Quanto, poi, alla difficoltà di programmazione vera e propria, è un aspetto che si può tranquillamente ignorare: provvederà il MacroAssembler Commodore a rendere tutto più semplice.

EL SID

Prima, però, è d'obbligo dare un'occhiata, anche se piuttosto superficiale, al "Sound Interface Device", per gli amici SID che non ha niente a che fare con i servizi segreti (battuta standard, risata facoltativa).

Si tratta, com'è noto, di un circuito integrato (il già citato 6581) dotato di tre oscillatori, nel linguaggio corrente definiti come "voci", ognuno dei quali può generare quattro forme d'onda: triangolare, a dente di sega, rumore, ed infine quella quadra, o, più corretamente, ad impulsi (l'onda quadra, infatti, è un particolare tipo di onda ad impulsi).

Poichè il nostro scopo è prettamente pratico, tralasciamo una più approfondita trattazione teorica sui tipi di onda sonora, rimandando il lettore alla manualistica varia, compresa l'utile Guida di Riferimento del Programmatore.

Per ora ci basti sapere che, modificando il tipo di onda sonora, si otterrà un diverso tipo di suono: ne constateremo tra breve le reali differenze.

Il SID può essere programmato, via software, manipolando opportunamente una serie di locazioni RAM, dette Registri, locate da 54272 a 54300, per un totale quindi di 28 byte.

Comunemente, ed anche noi ci atterremo a questa abitudine, ci si riferisce ai registri del SID definendoli col solo indice di posizione, per cui il registro 0 corrisponderà alla locazione 54272 + 0, il registro 1 a 54272 + 1 (= 54273), e così via.

Inoltre, come si può notare in figura 1, i primi tre gruppi di registri (numerati da 0 a 20) svolgono esattamente le stesse funzioni, ma con riferimento alle tre diverse voci.

Per tale motivo, tra l'altro, nei nostri esempi utilizzeremo solo la voce 1: quanto si vedrà, infatti, può essere applicato pari pari con le stesse modalità a-

MACROASSEMBLER E LIBRERIE

Nonostante si parli dell'assemblatore della Commodore sin dal n. 51, qualche nuova risorsa da esaminare continua a rendere più che interessante l'argomento.

E' questa la volta di due pseudo-istruzioni che possono veramente farci comodo...

.LIB
.FIL

...in cui il carattere iniziale di punto(.) è indispensabile per funzionare correttamente.

La loro utilizzazione consente di sfruttare routine personali, memorizzate su dischetto, che possono essere inserite nel programma che si sta elaborando: in altre parole, si può disporre di cosiddette "librerie" di programmi, alla stregua di sistemi ben più "robusti" del nostro C/64.

Proprio la routine proposta in queste pagine può fornirci uno spunto per adoperarle, e soprattutto capirle.

Il disassemblato in esame può, infatti, essere diviso in due parti: una principale, che si occupa di far suonare il C/64, ed una per così dire accessoria, ovvero la lista di dati riguardanti le note (righe da 65 in poi).

Un modo pratico per sfruttare al meglio una routine come questa, sarebbe quello di avere una copia del "corpo" principale del programma, e una serie di differenti sequenze musicali da aggiungere di volta in volta, a seconda delle nostre esigenze.

Ebbene, con .LIB e .FIL la cosa è resa possibile.

Basta inserire l'istruzione (in fase di editing) all'interno del programma principale, seguita da nome del file da aggiungere in quel punto.

Ovviamente, il file "esterno" dovrà essere anch'esso in formato MacroAssembler.

La differenza tra i due comandi è che con .LIB, in fase di assemblaggio, il Macroassembler lascia il programma e, dopo aver assemblato in quel punto il file esterno (deve trovarsi nello stesso dischetto!), torna al blocco principale per continuare il suo lavoro dallo stesso punto.

Con .FIL, invece, il nuovo file viene solo "agganciato" nel punto in cui viene menzionato, dopodichè l'assemblaggio si conclude.

Per esempio, la variazione sul tema proposta nell'articolo, può essere realizzata come segue:

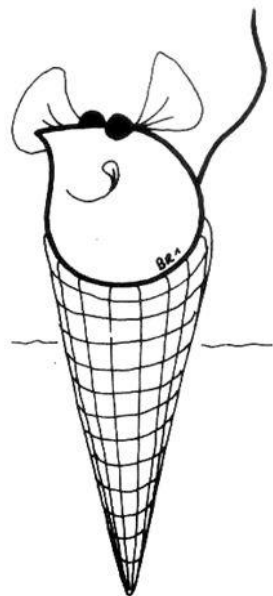
1) Si sostituisca la linea 69 con .lib altrenote
2) Con l'editor si crei un file sorgente così strutturato...

```
10      .byt 50, 10, 45
20      .byt 1, 0, 20
```

...e lo si salvi con Put"altrenote".

Tutto fatto.

Dopo l'assemblaggio (solo della routine principale!) ed il suo caricamento in memoria, il Re# e la pausa saranno stati aggiunti al codice macchina.

L'esempio proposto è, in effetti, alquanto semplice, ma, come forse si è capito, le possibili applicazioni di .LIB e .FIL sono veramente infinite, aggiungendo un ulteriore tocco di versatilità al nostro già "super" Macro Assembler.

PRINCIPALI REGISTRI DEL SID

indirizzo base = 54272

voce1	voce2	voce3	
Ø	7	14	FREQUENZA DELLA NOTA (byte basso)
1	8	15	FREQUENZA DELLA NOTA (byte alto)
2	9	16	AMPIEZZA FORMA D'ONDA (byte basso)
3	1Ø	17	AMPIEZZA FORMA D'ONDA (byte alto)
4	11	18	SELEZIONE FORMA D'ONDA
5	12	19	DURATA DI ATTACCO E DECADIMENTO
6	13	2Ø	DURATA DI SOSTEGNO E RILASCIO
24	24	24	VOLUME

FIGURA 1

La gestione dell'interrupt consente di ascoltare musica mentre gira un altro programma

gli altri due oscillatori, modificando opportunamente il solo numero di registro.

Del quarto gruppo di registri (da 21 a 28), in figura è riportato solo quello di controllo del volume, l'unico che, in questa sede, sarà utilizzato. Gli altri, molto più complessi da gestire, riguardano la filtratura e modificazione dinamica del suono durante la sua esecuzione.

Si tratta, in pratica, dei cosiddetti "effetti speciali", che al momento esulano dai nostri interessi (se ne riparlerà, magari, in futuro).

UN PO' DI TEORIA

Per ottenere un suono dall'amico SID, è necessario dunque settare, tramite i suddetti registri, un gruppo di parametri, e più precisamente:
1) La frequenza della nota desiderata.
2) Il tipo di onda del suono (triangolare, a dente di sega, ecc.).
3) L' ADSR.
4) Il volume.

Esaminiamo più in dettaglio (ma non troppo) le varie fasi, che vedremo tra poco applicate in una nostra routine.

La frequenza di una nota, ricavabile da apposite tabelle pubblicate nella manualistica, va inserita nei primi due registri (0 e 1) nel formato basso / alto.

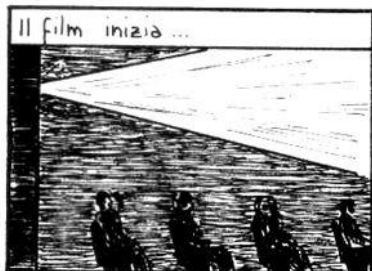
Operando in esadecimale, la cosa non presenta problemi; considerando per esempio il numero $D4A3, le prime due cifre (D4) sono il byte alto, le altre due (A3) quello basso.

In decimale, invece, come dovrebbe essere noto, definendo con NR il valore da scomporre, LO il byte basso ed HI il byte alto, si avrà:

HI = INT (NR / 256)
LO = NR - HI * 256

Il tipo di onda è selezionabile manipolando il regi-

CAP.BLITTER
by Paolini Andrea

VADO E LO DISTRUGGO...

REGISTRO DI SELEZIONE FORMA D'ONDA

bit	7	6	5	4	3	2	1	0
valore	128	64	32	16	8	4	2	1
	RUMORE	QUADRA	SEGA	TRIAN.	---	---	---	ON/OFF

FIGURA 2

stro 4 (per la voce 1), o meglio i suoi quattro bit più significativi, come illustrato in figura 2.

In pratica basta inserire il valore relativo al tipo di onda (128 per quella quadra, 64 per dente di sega, ecc.), tenendo presente che, qualora si scegliesse l'onda ad impulsi (bit 6), occorre settare anche la sua ampiezza tramite il registro 2 (byte basso) e il nibble basso del registro 3 (byte alto).

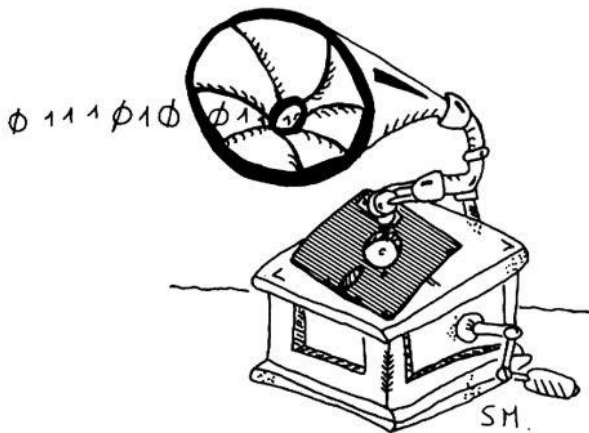
In effetti è necessario inserire il tipo di onda per ultimo, in quanto il bit 0 dello stesso registro (il cosiddetto Gate), se posto ad uno, provoca l'effettivo inizio di emissione della nota; in caso contrario possono verificarsi malfunzionamenti.

Dopo aver settato tutti i parametri, basterà pokare in questo registro il valore corrispondente al tipo di onda più uno (bit 0 settato), per far partire il suono; reinserendo il solo valore per il tipo di onda (bit 0 resettato), il suono cesserà.

Un'altra manipolazione indispensabile riguarda il generatore di inviluppo, ovvero la definizione dell'ADSR, che sta per Attacco, Decadimento, Sostegno e Rilascio.

Senza entrare troppo nei particolari teorici, con questi parametri vengono stabiliti i tempi (in millisecondi) di salita e ricaduta del segnale acustico, determinando la somiglianza del suono a quello di uno strumento piuttosto che ad un altro.

Anche qui, come nel caso precedente, a meno di disporre dei dati opportuni, è più semplice capire sperimentalmente come vanno le cose, ed è quello che faremo tra breve, ma un'occhiata ai registri ADSR (5 e 6) va comunque data.

Osservando le figure 3 e 4, si può notare come le quattro componenti dell'inviluppo siano raggruppate a coppie nei due registri che ne controllano la durata.

Questo significa che, per ognuna delle fasi, si può scegliere un valore che va da 0 a 15, valore massimo ottenibile con 4 bit (= nibble).

Per settare l' ADSR, quindi, occorrerà inserire un valore che comprenda tanto l'attacco quanto il decadimento nel registro 5 (= locazione 54277), e il sostegno e rilascio nel registro 6.

Più semplicemente, il valore da pokare può essere ricavato usando queste formulette:

Registro 5 = Att. * 16 + Dec.
Registro 6 = Sos. * 16 + Ril.

Sapevate che con il MacroAssembler Commodore è possibile gestire librerie?

PERSONAGGI ED INTERPRETI:

CAPITANO BLITTER	: LO SPRITE 1
IL VIRUS	: LO SPRITE 2
GLI AGENTI 32 & 64	: GLI SPRITE 3 e 4
IL RASTER	: SE STESSO
LA C.P.U.	: SE STESSA
IL FLOPPY	: UN FLOPPY DISK

SI RINGRAZIA LO SPRITE MULTICOLOR #6 CHE PER STAVOLTA HA LAVORATO IN MONOCOLOR.
SFONDO IN: KOALAPAINTER
MUSICHE DI: MICHELE MAGGI

REGIA DI: ANDREA PAOLINI

REGISTRO DI ATTACCO E DECADIMENTO

bit	7	6	5	4	3	2	1	0
valore	128	64	32	16	8	4	2	1
	ATT.3	ATT.2	ATT.1	ATT.0	DEC.3	DEC.2	DEC.1	DEC.0

FIGURA 3

Provate ad inserire un motivetto nell'interrupt

Il volume, infine, va regolato ponendo un numero da 0 a 15 nel registro 24 (locazione 54296).

Nella pratica, per questo parametro viene sempre adoperato il valore massimo... è più semplice modificare il volume del monitor (o TV).

Per concludere la carrellata di nozioni, è necessario ricordare anche un'altra componente del suono: la sua durata nel tempo.

E' questo un fattore molto importante, soprattutto quando si ha a che fare con la musica, e la sua gestione sarà chiarita tra breve.

FIATO ALLE TROMBE!

Vediamo, finalmente, di concretizzare quanto finora descritto, e possibilmente in modo tale da giustificare il ricorso al linguaggio macchina: con qualche poke (da Basic) nei registri appena descritti, infatti, è già possibile far emettere al computer suoni più o meno gradevoli, ma impegnandone tutte le risorse.

In altre parole, ricorrendo al Basic, è sì possibile gestire il suono con adeguati programmi infarciti di poke, data, e cicli di ritardo vari, ma non si potrà mai ottenere, tra l'altro, una musica da ascoltare mentre "gira" un altro programma.

Con l'aiuto del MacroAssembler, invece, non solo è possibile, ma risulta anche abbastanza semplice.

Lo possiamo vedere subito, cominciando col copiare il listato Basic di queste pagine, il cui compito è quello di installare in memoria (a partire da 49152) la routine LM di cui è pubblicato il disassemblato, in formato MacroAssembler.

REGISTRO DI SOSTEGNO E RILASCIO

bit	7	6	5	4	3	2	1	0
valore	128	64	32	16	8	4	2	1
	SOS.3	SOS.2	SOS.1	SOS.0	RIL.3	RIL.2	RIL.1	RIL.0

FIGURA 4

```
10 REM ---------------------------------------------------------
15 REM       GESTIONE DEL SUONO ATTRAVERSO L'INTERRUPT
20 REM ---------------------------------------------------------
25 :
30 FORX=0TO110:READA:B=B+A:POKE49152+X,A
35 NEXT:IFB=14299 THEN SYS 49152 : END
40 PRINT"ERRORE NELLE LINEE DATA!":END
45 :
50 DATA 120,169,045,162,192,141,020,003,142,021,003,162,001,134
55 DATA 251,202,142,002,212,169,008,141,003,212,169,160,141,005
60 DATA 212,169,250,141,006,212,169,015,141,024,212,169,067,133
65 DATA 252,088,096,198,251,208,045,166,253,189,097,192,208,014
70 DATA 232,189,097,192,208,007,170,168,200,132,251,208,023,202
75 DATA 141,000,212,232,189,097,192,141,001,212,232,189,097,192
80 DATA 133,251,165,252,141,004,212,232,134,253,076,049,234,156
85 DATA 013,030,070,015,030,032,012,030,021,009,030,000,000
90 END
```

Dopo il Run d'obbligo, e il breve intervallo necessario per allocare il codice macchina letto dalle linee Data, il computer tornerà alle sue condizioni normali, pronto cioè a ricevere comandi o ad eseguire programmi, ma con un (ossessivo) sottofondo sonoro che scandisce, in un loop infinito, quattro note musicali.

Certo, non è una sinfonia di Beethoven, ma per i nostri scopi è più che sufficiente.

LA ROUTINE

Come forse avrete già intuito, un simile risultato è possibile solo grazie all'uso dell'interrupt, la cui manipolazione è più chiaramente evidenziata nel disassemblato commentato.

Questo, o meglio la routine che rappresenta, può essere diviso in tre blocchi: il settore dalla riga 9 alla riga 31, che si limita a preparare i registri 2, 3, 5, 6 e 24 del SID e a dirottare il vettore di interrupt (le arcinote locazioni 788 e 789) in modo da farlo puntare alla sezione della nostra routine che inizia alla riga 33.

Da qui, fino alla linea di istruzioni 63, vengono prelevati i dati relativi alle note ed alla loro durata e, dopo averli inseriti nei registri 0 ed 1, si attiva l'emissione del suono ad essi relativo.

Infine, da riga 65 in poi, sono presenti, nell'ordine, le frequenze di ogni nota (prima il byte basso e poi quello alto) e la sua durata, proprio come se si trattasse di linee Data del Basic.

Ma entriamo nei dettagli, o almeno in quelli più significativi: quanto non descritto in queste righe, può essere facilmente desunto dai commenti posti nel disassemblato.

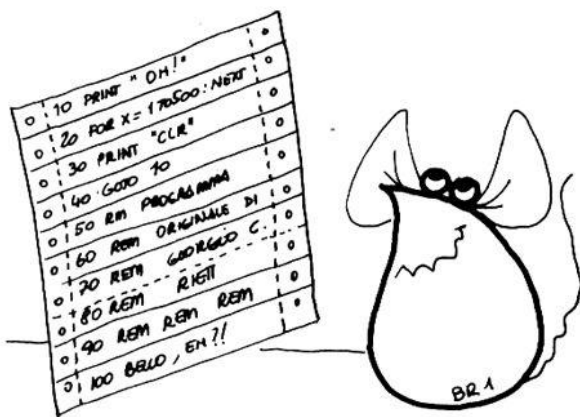
Niente di nuovo alle righe 9 - 13, l'argomento interrupt è stato più volte trattato anche nell'ambito del MacroAssembler.

In riga 15 si prepara la locazione 251, adibita a conservare, tra un ciclo di interrupt e l'altro, il tempo di durata di una nota.

Nelle istruzioni che seguono, viene impostato il tipo di onda quadrangolare (righe 27 - 28), e quindi viene necessariamente definita la sua ampiezza (17 - 20).

Per l'ADSR si inseriscono, nei relativi registri, valori piuttosto alti (il massimo è, ovviamente, 255), che in genere producono un suono tipicamente "elettronico", mentre il volume è regolato al massimo (25 - 26).

La routine vera e propria, che, si ricordi, viene ri-

chiamata ogni cinquantesimo di secondo, praticamente agisce così:

1) Decrementa il contenuto della locazione 251 (durata) e, finchè non ha raggiunto il valore 0, non esegue le istruzioni successive.
2) Legge il byte basso della frequenza della nota, prelevandolo dalle locazioni identificate dalla label NOTA, ed esegue un primo controllo per verificare se è uguale a zero (una sequenza di due zeri indica che si è giunti alla fine delle note da leggere). Si noti che per la lettura del dato viene utilizzato un indirizzamento indicizzato (37), con X che preleva dalla locazione 253 l'indice della posizione alla quale si è giunti.
3) Se il dato letto è uguale a zero, viene controllato se lo è anche il successivo (righe 40 - 47), nel qual caso viene posta ad 1 la durata (in modo che le note riinizino subito dopo il DEC di riga 33) e riazzerato l'indice di lettura delle note, dopodichè si esce al normale indirizzo di interrupt del sistema.
4) Se il dato di cui al punto 2 non è risultato zero, si procede a leggere il byte alto della nota (52 - 53), si inseriscono i valori della frequenza nei registri 0 ed 1 del SID (righe 50 e 53), e si attiva la nota tramite il bit 0 del registro 4 di selezione (58).
5) Si legge la durata della nota (55) e si deposita questo valore nella locazione 251, quindi si salva anche il flag (indice) di posizione delle note, e si esce.

Va notato che, per com'è strutturata, la routine consente il trattamento di un massimo di 256 dati relativi alle note, dato che per il flag-indice viene utilizzata la sola locazione 253.

Volendo potenziare questa caratteristica, basterà (si fa per dire!) usare due locazioni invece di una e, soprattutto, ricorrere all'indirizzamento indiretto postindicizzato per leggere e "storare" i dati.

A voi il piacere di realizzarlo.

VARIAZIONI SUL TEMA

La routine proposta, comunque, si presta anche a modifiche meno drastiche.

E' possibile, infatti, mentre è in esecuzione, variarne alcuni parametri in modo da sperimentare nuove soluzioni sonore.

Se qualcosa dovesse andar storto, basta premere Run / Stop + Restore (nei casi più "ostinati", resettare il sistema col tastino o con sys 64738) e ridare Sys 49152.

Per esempio, si può modificare il tipo di onda sonora, semplicemente modificando il contenuto della locazione 252: provate con Poke 252, x (con x che può assumere i valori 129, 65, 33 e 17), e potrà essere realmente apprezzata la differenza tra i vari tipi.

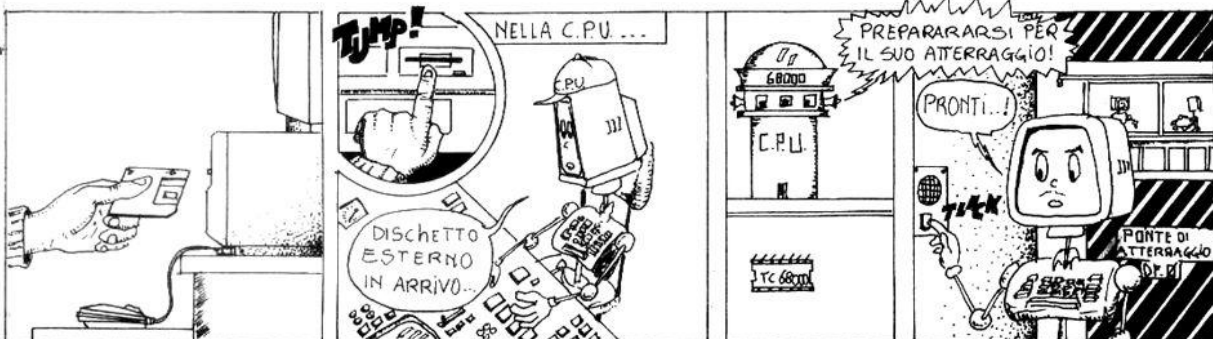
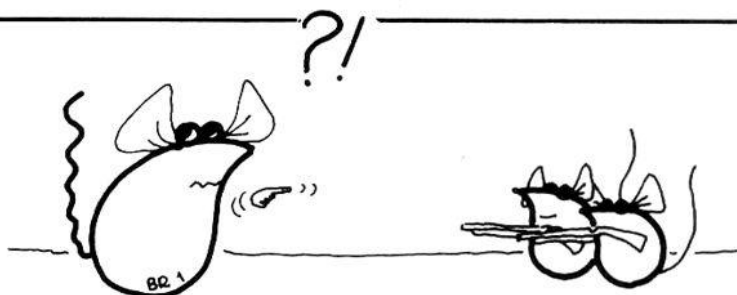
Si provi anche a modificare direttamente, sempre con delle Poke, il contenuto dei registri 5 e 6 (locazioni 54277 e 54278), e, perchè no, anche le note e la loro durata.

Come esempio, si digitino le seguenti righe in Basic (sempre con la routine che continua a "suonare")...

```
10 FOR X=0 TO 7: READ A
20 POKE 49261 + X, A
30 NEXT: END
40 DATA 50, 10, 45
50 DATA 01, 00, 20
60 DATA 00, 00
```

...che aggiungono, in coda ai dati già presenti, un'altra nota (Re#) e una pausa, ottenuta inserendo una frequenza non udibile 1 (0, 0 non può essere usato per ovvi motivi) e durata 20 (Vedi anche riquadro).

Dopo il Run, sarà... tutta un'altra musica: con un po' di fantasia, dovrebbe ricordare la sequenza di "Incontri Ravvicinati del terzo tipo", quella che dimostrava agli alieni l'intelligenza di noi terrestri.

Beh, la situazione in fondo potrebbe essere la stessa, solo che a lanciare il segnale è il nostro C/64...

DISASSEMBLATO COMMENTATO

```
01 IRQVET =$314      ;dec.788
02 IRQSYS =$EA31     ;dec.59953.
03 REG    =$D400     ;dec.54272
04 DURATA =$FB       ;dec.251
05 ONDA   =$FC       ;dec.252
06 FLAGNT =$FD       ;dec.253
07        *=$C000    ;start=49152
08 ;---------------------------------
09        SEI          ;Disabilita IRQ.
10        LDA #<ROUT   ;Byte basso e
11        LDX #>ROUT   ;byte alto di
12        STA IRQVET   ;Rout nel vettore
13        STX IRQVET+1 ;di interrupt.
14 ;---------------------------------
15        LDX #1       ;Prepara primo
16        STX DURATA   ;start del suono.
17        DEX          ;Ampiezza / low
18        STX REG+2    ;per onda quadra.
19        LDA #8       ;Ampiezza / hi
20        STA REG+3    ;per onda quadra.
21        LDA #160     ;Attacco e
22        STA REG+5    ;decadimento.
23        LDA #250     ;Sostegno e
24        STA REG+6    ;rilascio.
25        LDA #15      ;Volume impostato
26        STA REG+24   ;al massimo.
27        LDA #65      ;Forma d'onda
28        STA ONDA     ;quadrangolare.
29 ;---------------------------------
30        CLI          ;Riabilita IRQ.
31        RTS          ;Return al basic.
32 ;---------------------------------
33 ROUT   DEC DURATA   ;Determina tempo
34        BNE USCITA   ;della nota.
35 ;---------------------------------
36        LDX FLAGNT   ;X = Flag nota.
37        LDA NOTA,X   ;Legge nota (low).
38        BNE CONT     ;Salta se <> 0.
39 ;---------------------------------
40        INX          ;X = X + 1.
41        LDA NOTA,X   ;Legge nota (hi).
42        BNE DECR     ;Salta se <> 0.
43        TAX          ;X = 0.
44        TAY          ;Y = 0.
45        INY          ;Y = Y + 1.
46        STY DURATA   ;Resetta durata.
47        BNE STOREX   ;Salta (JMP).
48 ;---------------------------------
49 DECR   DEX          ;X = X - 1.
50 CONT   STA REG      ;Frequenza (low).
51        INX          ;X = X + 1.
52        LDA NOTA,X   ;Legge e deposita
53        STA REG+1    ;frequenza (hi).
54        INX          ;X = X + 1.
55        LDA NOTA,X   ;Setta la durata
56        STA DURATA   ;della nota.
57 ;---------------------------------
58        LDA ONDA     ;Attiva la forma
59        STA REG+4    ;d'onda scelta.
60        INX          ;X = X + 1.
61 STOREX STX FLAGNT   ;Salva flag nota.
62 ;---------------------------------
63 USCITA JMP IRQSYS   ;IRQ di sistema.
64 ;---------------------------------
65 NOTA   .BYT 156,13,30 ; Sol#.
66        .BYT 70,15,30  ; La #.
67        .BYT 32,12,30  ; Fa #.
68        .BYT 21,9,30   ; Do #.
69        .BYT 0,0       ; Fine note.
70        .END           ;Fine assemblaggio.
```

ABBATTETE LE BARRIERE!

Come aggirare la più diffusa tecnica di autorun su disco

di **Gianluca Venturini**

Una tecnica semplice consente di aggirare l'ostacolo della protezione

Già ampiamente si è parlato sulle pagine di C.C.C. di protezioni. In particolare, le tecniche di autorun proposte agiscono, generalmente, nello stesso modo: creano una routine che non solo disabilita errori, tasti run / stop, restore e reset ma, cosa più importante, manda in esecuzione il programma.

Le routine di cui parliamo creano, a loro volta, un'altra routine che, una volta caricato il programma da proteggere, lo salvano assieme all'area contenente i vettori di sistema, indirizzando quello di Warm Start alla ruotine precedente.

In questo modo, come già saprete, caricando il file con...

Load " nomefile", 8, 1

...si ha l'immediata esecuzione al termine del load. Se provassimo a caricare il file con il solo suffisso " .8" otterremmo in memoria un file senza senso... così almeno sembra.

IL TRUCCO

Listando, infatti, un programma dotato di autostart, caricato nel modo sopra descritto (cioè con il semplice suffisso " ,8"), si nota, sullo schermo, la comparsa di simboli alfanumerici che, con il programma vero e proprio, non hanno nulla a che fare.

Il programma, tuttavia, è presente in memoria e, addirittura, listabile; solo che... si trova un po' più in là del solito.

Spieghiamoci meglio con un esempio che, essendo un po' contorto, è bene leggere con la massima attenzione.

Un programma Basic (dotato ovviamente di autorun) che inizia dalla locazione 770, caricato con il solo suffisso " ,8" viene allocato a partire dalla locazione 2049, ma, nonostante ciò, continua ad esser costituito dagli stessi bytes che dovrebbero venir caricati (",8,1") a partire da 770.

Pertanto i primi (2049 - 770 =) 1279 bytes a partire da 2049 sono quelli che prima venivano caricati da 770 a 2048. I bytes del programma vero e proprio, che prima venivano caricati da 2049 in poi, ora si trovano da (2049 + 1279 =) 3328 in poi.

Per visualizzare il programma Basic originale, " scavalcando" l'autorun, basterà trasferire i bytes 3328 (e successivi) nelle locazioni a partire da 2049 in poi.

IL PROGRAMMA

Il listato pubblicato in queste pagine richiede, in input, il nome del file, presente su disco, da " sproteggere".

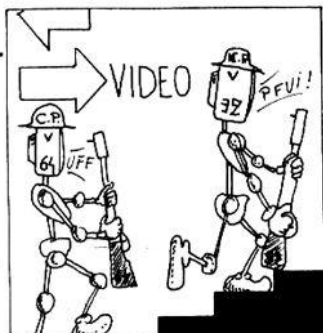
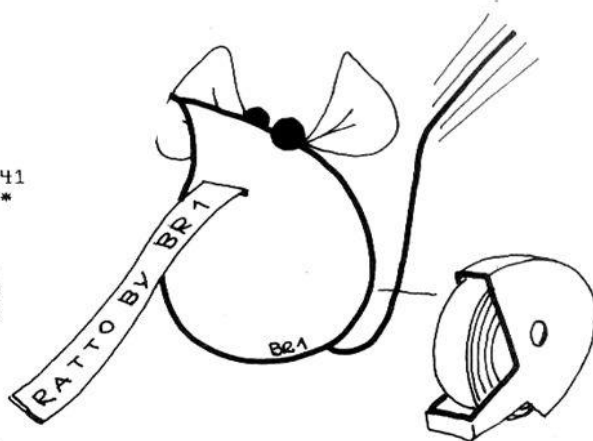
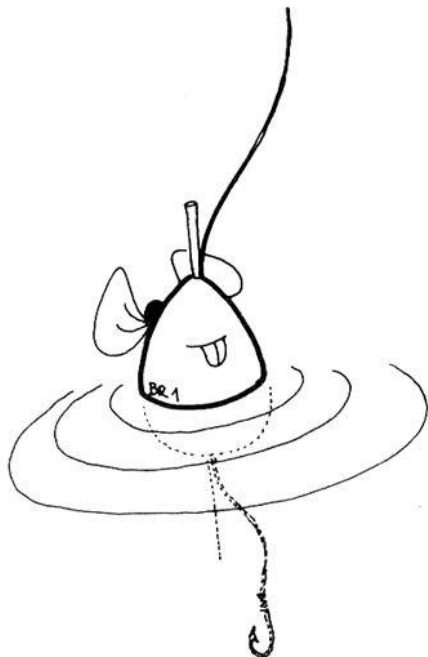
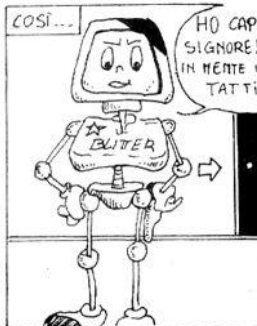
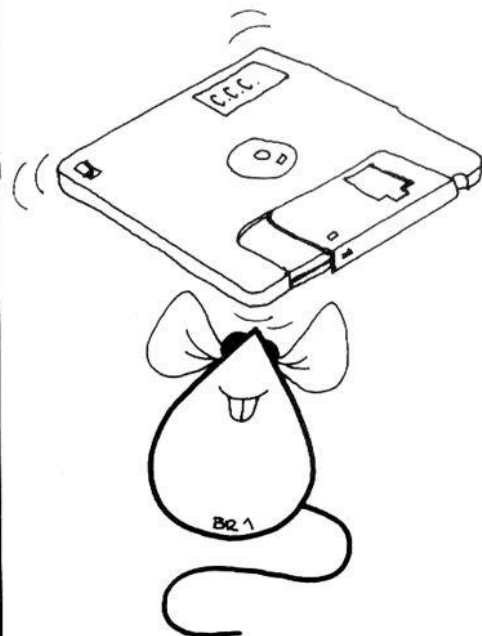
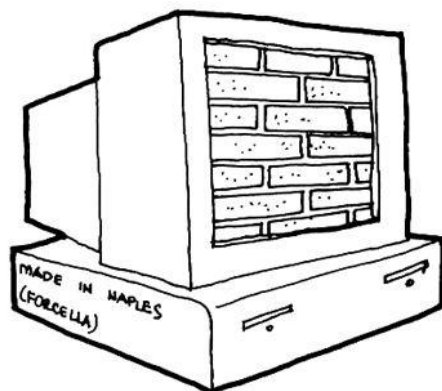
Dal momento che, per semplicità, il programma non effettua controlli, se vengono ricontrati erroridi di I/O rilanciate " Anti-Autorun".

Per i files da sproteggere è doverosa un'avvertenza: il file deve essere lungo abbastanza da interessare anche l'area Basic in modo da essere esaminato. Il procedimento non funziona con i sistemi del tipo Loader Isepic, lunghi pochissi blocchi.

Viene quindi rilevata la locazione iniziale del file, indicata dai suoi primi due bytes (linee 20 - 60); poi, mediante, il calcolo 2049-start, sapremo il numero di locazioni da " escludere", a partire da 2049, per trovare il file che interessa (differenza; linee 80-100).

Sommando quindi la differenza a 2049 conosceremo l'inizio del programma caricando il file con il solo suffisso ",8".

Viene quindi caricata in memoria, e lanciata, una routine L.M. che svolge diversi compiti:

```
1 REM ANTI AUTORUN PER CBM 64 + 1541
2 REM ****BY GIANLUCA VENTURINI****
3 REM      N.B.: IL PROGRAMMA DA
4 REM    SPROTEGGERE DEVE ESSERE
6 REM  ABBASTANZA LUNGO DA OCCUPA-
7 REM  RE ALMENO LA LOCAZIONE 2049
8 REM  (NO AD ESEMPIO CAR. ISEPIC)
9 REM
10 INPUT"NOME PROGRAMMA";A$
20 OPEN3,8,3,A$:GET#3,B$,C$
30 IFB$=""THENB$=CHR$(0)
40 IFC$=""THENC$=CHR$(0)
50 POKE251,ASC(B$):POKE252,ASC(C$)
60 PRINT"{CLR}START: "PEEK(251)+256*PEEK(252)
70 CLOSE3:IFPEEK(252)*256+PEEK(251)>2049THENPRINT"FILE NON AUTORUN":END
80 POKE252,8-PEEK(252):R=1-PEEK(251):IFR<0THENR=RAND255:POKE252,PEEK(252)-1
90 POKE251,R
100 PRINT"DIFFERENZA: "PEEK(251)+256*PEEK(252)
110 A=PEEK(251)+1:IFA=256THENA=0:POKE252,PEEK(252)+1
115 POKE251,A:POKE252,PEEK(252)+8
120 PRINT"INIZIO PRG: "PEEK(251)+256*PEEK(252)
125 PRINT"PROGRAMMA ALLOCATO DA 2049 A ";:REM PUNTO E VIRGOLA
130 POKE139,1:POKE140,8:POKE253,LEN(A$)
140 FORI=0TO150:READA:POKE828+I,A:NEXT
150 SYS828
200 DATA 165,253,166,051,232,232,164,052,032,189,255,169,000
210 DATA 162,008,160,000,032,186,255,169,000,160,008,162,001
220 DATA 032,213,255,134,141;132,142,120,169,053,133,001,238
230 DATA 032,208,160,000,177,251,145,139,230,139,208,002,230
240 DATA 140,230,251,208,002,230,252,165,252,197,142,208,229
250 DATA 165,251,197,141,208,223,169,055,133,001,088,165,140
260 DATA 166,139,202,224,255,208,003,056,233,001,032,205,189
270 DATA 169,013,032,210,255,160,003,200,177,043,208,251,200
280 DATA 200,152,160,000,145,043,165,044,200,145,043,133,060
290 DATA 160,000,132,059,162,000,200,208,002,230,060,177,059
300 DATA 208,245,232,224,003,208,242,200,208,002,230,060,132
310 DATA 045,164,060,132,046,108,002,003
320 END
```

```
DISASSEMBLATO
COMMENTATO

Routine:ANTI-AUTORUN
By Gianluca Venturini
Origin:$033C

       LDA $FD    ;lunghezza nome file
       LDX $33
       INX
       INX          ;#<filename in X
       LDY $34      ;#>filename in Y
       JSR $FFBD    ;SETNAM
       LDA #$00
       LDX #$08     ;drive
       LDY #$00     ;rilocazione
       JSR $FFBA    ;SETLFS
       LDA #$00     ;0=load
       LDY #$08     ;caricamento a
       LDX #$01     ;partire da 2049
       JSR $FFD5    ;LOAD
       STX $8D      ;salva LO e HI byte del-
       STY $8E      ;l'ultima locazione caricata
       SEI
       LDA #$35
       STA $01
*0362  INC $D020    ;cambia colore al bordo
       LDY #$00     ;sposta
       LDA ($FB),Y  ;  il
       STA ($8B),Y  ;programma
       INC $8B      ;  da
       BNE *0371    ;'inizio prg.'
       INC $8C      ;  a
*0371  INC $FB      ;2049
       BNE *0377    ;  e
       INC $FC      ;seguenti
*0377  LDA $FC
       CMP $8E
       BNE *0362
       LDA $FB
       CMP $8D
       BNE *0362
       LDA #$37
```

```
        STA $01
        CLI
        LDA $8C
        LDX $8B
        DEX
        CPX #$FF
        BNE *0394
        SEC
        SBC #$01
*0394   JSR $BDCD    ;stampa l'ultima
        LDA #$0D     ;locazione occupata
        JSR $FFD2    ;e un Carriage Return
        LDY #$03     ;rimette a posto
*039E   INY          ;i puntatori
        LDA ($2B),Y
        BNE *039E
        INY
        INY
        TYA
        LDY #$00
        STA ($2B),Y
        LDA $2C
        INY
        STA ($2B),Y
        STA $3C
        LDY #$00
        STY #$3B
*03B5   LDX #$00
*03B7   INY
        BNE *03BC
        INC $3C
*03BC   LDA ($3B),Y
        BNE *03B5
        INX
        CPX #$03
        BNE *03B7
        INY
        BNE *03CA
        INC $3C
*03CA   STY $2D
        LDY $3C
        STY $2E
        JMP ($0302) ;torna in modo diretto
```

Non sempre la procedura funziona: ma tentar non nuoce!

- Carica il file da 2049;
- Riporta il programma da 2049 in poi;
- Stampa l'ultimo byte occupato dal programma;
- " Aggiusta" i puntatori (la routine è tratta da " COMMODORE" n. 3).

Per memorizzare ora il programma così caricato, oltre al Save (inefficace se vi sono parti in L.M.), suggeriamo la seguente procedura:

- digitare Sys 57812 " Nomefile", 8, 1
- digitare Poke 193, 1: Poke 194, 8: Poke 174, Peek (139): Poke 175, Peek (140)
- Digitare Sys 62954.

LIMITAZIONI

Esistono programmi lanciati da un loader che, per forza di cose, deve essere dotato di autostart.

In questo caso caricare con Anti-Autorun il loader e modificare la parte della routine che provoca l'esecuzione sostituendola con istruzioni che restituiscano il controllo all'utente (esempio: RTS).

Nel caso di loader che lavorino su routines che non occupino l'area da 2049, può essere valido il seguente procedimento (se siete bravi), valido anche per autorun con crittografia:

- caricate il file mediante il solo suffisso " ,8"
- esaminate, servendovi di un disassemblatore, le locazioni da 2049 al fine di capire come agisce la protezione o la crittografia;
- modificare le routines in modo da non provocare l'esecuzione del programma;
- riportare le stesse routines nella loro area originaria (servendosi delle informazioni di testa di Anti-Autorun) senza modificare i wedges;
- eseguire le routines se si tratta di un loader;
- riportare con delle Poke X, Peek (Y) in un ciclo FOR..NEXT il programma da 2049 (e seguenti) quindi...
- impartire la SYS della routine.

Il discorso è probabilmente poco chiaro ai meno esperti, anche perchè non sempre è facile l'attuazione pratica di quanto detto.

Noi stessi, comunque, effettuando vari tentativi (tra cui quelli descritti) riusciamo quasi sempre nell'intento...

COME GESTIRE 64 SPRITE

Visualizziamo 64 sprites, tutti diversi tra loro, sul nostro C/64

di **Gianluca Venturini**

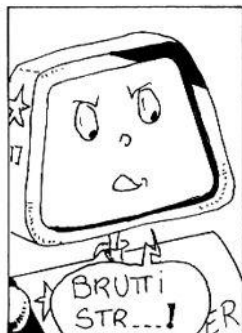
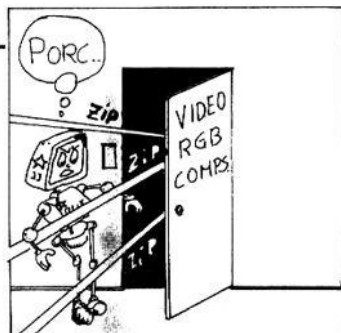
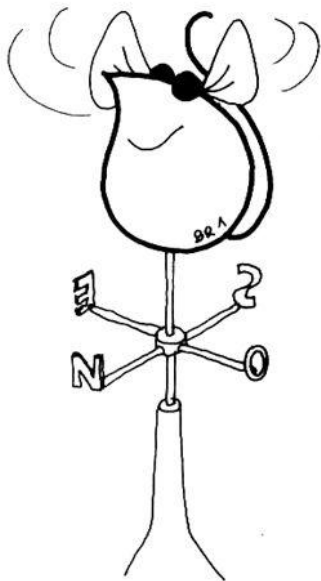
Il programma rappresenta una versione migliorata del listato pubblicato in occasione della pubblicazione dell'articolo " La moltiplicazione degli sprites" (C.C.C. n. 50, pag. 69 e segg.).

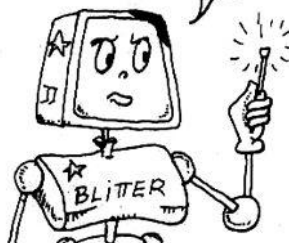
La routine L.M. (cui segue disassemblato) è sostanzialmente eguale alla precedente solo per quanto riguarda la gestione del raster. In aggiunta, per ottenere l'effetto desiderato (cioè 64 sprites diversi tra loro), vengono cambiati, ogni volta che la routine disegna un gruppo di 8 sprites, i puntatori 2040/7.

L'area visualizzata dai 64 sprites è quella compresa tra 12288 e 16383 che deve, pertanto, risultare libera da programmi Basic e/o linguaggio macchina (attenti ai puntatori di fine Basic!).

In pratica, quando vengono disegnati i primi otto sprites, le locazioni da 2040 a 2047 contengono i valori 192/3/4/5/6/7/8/9: quando verranno disegnati gli ultimi 8 sprites i puntatori 2040/7 conterranno, invece, i valori da 248 a 255. Il tutto in interrupt.

Per una comprensione più dettagliata dell'argomento si consiglia di rileggere l'articolo relativo al programma citato.

```
DISASSEMBLATO:

  1 *****************************
  2 *
  3 * UTILIZZO DEL RASTER PER
  4 *
  5 * DISEGNARE 64 SPRITES
  6 *
  7 * DIVERSI TRA LORO
  8 *
  9 * BY GIANLUCA VENTURINI 1989
 10 *
 11 *****************************
 12           ORG   49152
 13           JMP   START
 14 VISUAL    STX   $D001
 15           STX   $D003
 16           STX   $D005
 17           STX   $D007
 18           STX   $D009
 19           STX   $D00B
 20           STX   $D00D
 21           STX   $D00F
 22           STY   $07F8
 23           INY
 24           STY   $07F9
 25           INY
 26           STY   $07FA
 27           INY
 28           STY   $07FB
 29           INY
 30           STY   $07FC
 31           INY
 32           STY   $07FD
 33           INY
 34           STY   $07FE
 35           INY
 36           STY   $07FF
 37           INY
 38           RTS
 39 START     LDA   #$7F
 40           STA   $DCOD
 41           LDA   #<INTERR
 42           STA   $0314
 43           LDA   #>INTERR
 44           STA   $0315
 45           LDA   $D011
 46           AND   #$7F
```

```
47          STA   $D011      88          LDX   #$7E
48          LDA   #$81       89          LDY   #$D8
49          STA   $D01A      90          JSR   VISUAL
50          NOP              91          LDA   #$92
51          RTS              92          STA   RASTER
52 INTERR   NOP              93          JMP   EXIRQ
53          LDA   $D019      94 SPR5     LDA   RASTER
54          AND   #$01       95          CMP   #$97
55          BNE   SPR1       96          BCS   SPR6
56          JMP   EXIRQ      97          LDX   #$97
57 SPR1     STA   $D019      98          LDY   #$E0
58          LDA   RASTER     99          JSR   VISUAL
59          CMP   #$33      100          LDA   #$AB
60          BCS   SPR2      101          STA   RASTER
61          LDX   #$33      102          JMP   EXIRQ
62          LDY   #$C0      103 SPR6     LDA   RASTER
63          JSR   VISUAL    104          CMP   #$B0
64          LDA   #$47      105          BCS   SPR7
65          STA   RASTER    106          LDX   #$B0
66          JMP   EXIRQ     107          LDY   #$E8
67 SPR2     LDA   RASTER    108          JSR   VISUAL
68          CMP   #$4C      109          LDA   #$C4
69          BCS   SPR3      110          STA   RASTER
70          LDX   #$4C      111          JMP   EXIRQ
71          LDY   #$C8      112 SPR7     LDA   RASTER
72          JSR   VISUAL    113          CMP   #$C9
73          LDA   #$60      114          BCS   SPR8
74          STA   RASTER    115          LDX   #$C9
75          JMP   EXIRQ     116          LDY   #$F0
76 SPR3     LDA   RASTER    117          JSR   VISUAL
77          CMP   #$65      118          LDA   #$DD
78          BCS   SPR4      119          STA   RASTER
79          LDX   #$65      120          JMP   EXIRQ
80          LDY   #$D0      121 SPR8     LDX   #$E3
81          JSR   VISUAL    122          LDY   #$F8
82          LDA   #$79      123          JSR   VISUAL
83          STA   RASTER    124          LDA   #$00
84          JMP   EXIRQ     125          STA   RASTER
85 SPR4     LDA   RASTER    126          JMP   IRQ
86          CMP   #$7E      127 **********************
87          BCS   SPR5      128 RASTER   =     $D012
                            129 EXIRQ    =     $FEBC
                            130 IRQ      =     $EA31
                            131 **********************
```

```
0 REM ***64 SPRITES DIFFERENTI***
1 REM **BANCHI SPRITES 192..255**
2 REM ******(12288...16383)******
3 REM *(W) GIANLUCA VENTURINI 89*
4 REM
10 CK-0:FORI-0TO272:READA:POKE49152+I,A:CK-CK+A:NEXT
20 IFCK<>36390 THEN PRINT"ERRORE NEI DATA":END
30 PRINTCHR$(147)CHR$(5):REM *DEMO*
40 PRINT"ORA RIEMPO UNO PER UNO I 64 SPRITE"
41 PRINT"(PREMI UN TASTO)"
42 GETQQ$:IFQQ$-""THEN42
45 J-0:POKE53275,0
50 POKE53248+J*2,30+28*J:POKE53287+J,0:J-J+1:IFJ<8THEN50
60 POKE53269,255:SYS 49152
70 FORI-12288TO16383:PRINT CHR$(19)TAB(34)I
75 POKEI,127:NEXT:REM   *****RIEMPIMENTO*****
80 POKE53275,255:END
100 DATA 076,060,192,142,001,208,142,003,208,142
110 DATA 005,208,142,007,208,142,009,208,142,011
120 DATA 208,142,013,208,142,015,208,140,248,007
130 DATA 200,140,249,007,200,140,250,007,200,140
140 DATA 251,007,200,140,252,007,200,140,253,007
150 DATA 200,140,254,007,200,140,255,007,200,096
160 DATA 169,127,141,013,220,169,090,141,020,003
170 DATA 169,192,141,021,003,173,017,208,041,127
180 DATA 141,017,208,169,129,141,026,208,234,096
190 DATA 234,173,025,208,041,001,208,003,076,188
200 DATA 254,141,025,208,173,018,208,201,051,176
210 DATA 015,162,051,160,192,032,003,192,169,071
220 DATA 141,018,208,076,188,254,173,018,208,201
230 DATA 076,176,015,162,076,160,200,032,003,192
240 DATA 169,096,141,018,208,076,188,254,173,018
250 DATA 208,201,101,176,015,162,101,160,208,032
260 DATA 003,192,169,121,141,018,208,076,188,254
270 DATA 173,018,208,201,126,176,015,162,126,160
280 DATA 216,032,003,192,169,146,141,018,208,076
290 DATA 188,254,173,018,208,201,151,176,015,162
300 DATA 151,160,224,032,003,192,169,171,141,018
310 DATA 208,076,188,254,173,018,208,201,176,176
320 DATA 015,162,176,160,232,032,003,192,169,196
330 DATA 141,018,208,076,188,254,173,018,208,201
340 DATA 201,176,015,162,201,160,240,032,003,192
350 DATA 169,221,141,01A,208,076,188,254,162,227
360 DATA 160,248,032,003,192,169,000,141,018,208
370 DATA 076,049,234
380 END
```

UNA CORSA IN VIDEOMOBILE

Spostiamo dove ci pare, con il joystick, qualunque porzione dello schermo low-res; e non solo questo...

di **Domenico Pavone**

Stack; come usare le locazioni di pagina 1

La gestione di parziali aree dello schermo, le cosiddette finestre (o, all'inglese, "window"), è uno di quegli argomenti che, ciclicamente, sembrano destare un notevole interesse nella fascia di aspiranti programmatori che dispongono di un "misero" C/64.

In effetti, nelle successive evoluzioni del non più giovane Basic 2.0, è quasi sempre presente un comando Window, adatto alla bisogna.

Non si creda, però, che siano tutte rose e fiori: nelle versioni che implementano questa facilitazione, è sì possibile redirigere l'output dello schermo in una finestra predefinita, ma la cosa finisce lì.

Sempre meglio che niente, direte, ma non è il caso di rodersi dall'invidia.

Intanto, anche sul nostro computer è possibile emulare il comando Window, come più volte è stato proposto sulla rivista.

Per applicazioni più sofisticate, la differenza tra il C/64 ed i suoi più illustri colleghi, si assottiglia decisamente: è necessario, indipendentemente dal tipo di computer, ricorrere ad una buona dose di programmazione in linguaggio macchina, e con risultati che non farebbero certo sfigurare il nostro beniamino.

Domanda scontata: "Anche nei confronti di un Amiga?"

Risposta, altrettanto scontata: "Non esageriamo..."

Vediamo comunque cosa si può tirar fuori dal sempiterno 64, con l'aiuto dell'ormai consueto MacroAssembler, cogliendo l'occasione per un utile ripasso di già affrontate istruzioni dell'Assembly.

Dulcis in fundo, analizzeremo come sfruttare una delle più importanti risorse del 6510: lo Stack.

GIROVAGANDO PER IL VIDEO

A beneficio anche di chi non è interessato all'aspetto didattico (peggio per lui!), cominciamo subito col vedere cosa fa, e come va utilizzata, la routine che tra breve esamineremo più a fondo.

Anzitutto, prima e dolorosa incombenza, occorre copiare il listato "caricatore" di queste pagine, ben zeppo di linee Data contenenti il codice macchina, che viene allocato in memoria a partire dalla tradizionale locazione 49152.

Vale la solita raccomandazione: salvare il programma PRIMA di mandarlo in esecuzione: eventuali errori di copiatura potrebbero farvi perdere tutto il lavoro fatto.

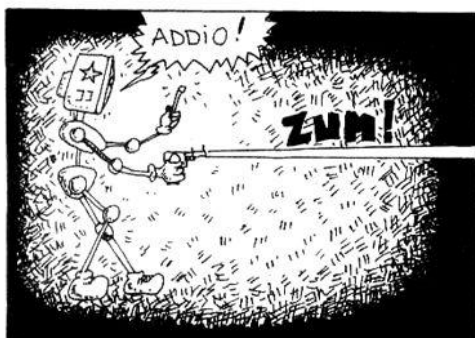
Ultimata la fatica, non resta che inserire un joystick in porta 2 ed impartire Run.

Dopo una breve attesa, riapparirà il Ready, preceduto da brevi indicazioni sulla sintassi da seguire per attivare il programma LM:

SYS 49152, X1, Y1, X2, Y2

... con l'obbligo di inserire le virgole (pena un Syntax Error), mentre i parametri che seguono la Sys servono per delimitare i confini di una finestra video.

Più precisamente:

• X1 indica il numero di riga dell'angolo superiore sinistro della finestra e, dato che ci si riferisce allo schermo in bassa risoluzione, dovrà essere compreso tra 1 e 25.

```
100 REM -------------------------------------------------------
110 REM  F I N E S T R A   M O B I L E   I N   L O W - R E S
120 REM              PER C64 E JOYSTICK IN PORTA 2
130 REM -------------------------------------------------------
140 PRINT CHR$ (147):PRINT"ATTENDI UN MOMENTO..."
150 FORX=0TO353:READA:B=B+A:POKE49152+X,A:NEXT
160 IFB<>49167THENPRINT"ERRORE NELLE LINEE DATA!":END
170 PRINTCHR$(147)"ATTIVAZIONE: SYS49152,X1,Y1,X2,Y2"
180 PRINT:PRINT"X1 = RIGA ANGOLO SUP. SINISTRO (1-25)"
190 PRINT "Y1 = COLONNA ANGOLO SUP.SINISTRO (1-40)"
200 PRINT "X2 - NUMERO DI RIGHE DELLA FINESTRA"
210 PRINT "Y2 - NUMERO DI COLONNE DELLA FINESTRA"
220 END
230 DATA 032,000,226,134,247,032,000,226,202,134,248,032,000,226
240 DATA 134,249,032,000,226,134,250,138,101,248,201,042,176,010
250 DATA 133,165,165,247,101,249,201,027,144,003,076,072,178,133
260 DATA 002,169,000,133,251,133,253,133,169,032,009,193,169,200
270 DATA 133,170,166,247,169,004,133,168,165,248,133,167,202,240
280 DATA 009,024,105,040,144,246,230,168,208,242,072,165,168,072
290 DATA 032,048,193,104,133,170,104,133,169,032,089,193,120,169
300 DATA 111,141,020,003,169,192,141,021,003,088,076,174,167,165
310 DATA 203,201,004,208,007,162,097,142,161,192,208,036,201,005
320 DATA 208,007,162,049,142,161,192,208,025,173,000,220,201,127
330 DATA 240,018,074,144,018,074,144,035,074,144,052,074,144,069
340 DATA 074,176,003,032,009,193,076,049,234,166,002,202,228,249
350 DATA 240,246,134,002,056,165,169,233,040,176,062,198,170,208
360 DATA 058,166,002,232,224,027,240,226,134,002,024,165,169,105
370 DATA 040,144,042,230,170,208,038,166,165,202,228,250,240,206
380 DATA 134,165,056,165,169,233,001,176,022,198,170,208,018,166
390 DATA 165,232,224,042,176,186,134,165,024,165,169,105,001,144
400 DATA 002,230,170,133,169,072,165,170,072,032,017,193,032,048
410 DATA 193,104,133,170,104,133,169,032,089,193,076,160,192,169
420 DATA 004,133,252,169,196,133,254,162,004,160,000,177,251,145
430 DATA 253,200,208,249,230,252,230,254,202,208,242,134,251,134
440 DATA 253,169,196,162,004,133,252,134,254,096,166,249,160,000
450 DATA 177,167,145,169,200,196,250,208,247,202,240,024,024,165
460 DATA 167,105,040,133,167,144,002,230,168,024,165,169,105,040
470 DATA 133,169,144,222,230,170,208,218,096,169,000,133,167,169
480 DATA 200,133,168,096
490 END
```

Un joystick è necessario per provare questa straordinaria routine

• Y1 identifica la colonna di inizio, compresa tra i valori 1 e 40.
• Con X2 ed Y2 vanno invece specificate le dimensioni della finestra, vale a dire, nell'ordine, di quante righe (X2 = altezza) e di quante colonne (Y2 = larghezza) deve essere composta.

Qualora si inserissero valori che eccedono le dimensioni dello schermo, si otterrebbe solo un Illegal Quantity Error.

La sys, per la gioia dei "basicofili", può essere impartita sia in modo diretto che da programma, per cui possiamo collaudare la routine digitando, dopo un opportuno New (e Return), queste poche righe...

```
10 print chr$(147);: for y=1 to 3
20 for x=1 to 3: print chr$(209);
30 next: print: next
40 print "muovi il joystick"
50 sys 49152, 1, 1, 3, 3
```

...che definiscono una piccola finestra (3 x 3), composta da pallini pieni.

Agendo sulla leva del joystick, si vedrà come questa può essere spostata liberamente all'interno del perimetro dello schermo, senza mai superarlo, e, soprattutto, senza che il suo contenuto vada perso (la cosa può essere facilmente rilevata facendo passare la finestra mobile sulla scritta).

Se si considera la scioltezza e velocità del movimento della window, già questo potrebbe considerarsi un buon risultato; ma c'è dell'altro.

Come si può dedurre dalla presenza del cursore, il computer non viene impegnato totalmente dalla routine, anzi rimane disponibile per qualunque altra normale attività... compresa quella di disegnare utilizzando i caratteri semigrafici.

Se però ci si limita a digitare qualcosa e poi si muove la leva del joystick, per motivi che saranno più chiari tra breve, riappare lo schermo precedente, eliminando quanto appena elaborato.

Il rimedio c'è: basta premere il pulsante di fuoco del joy, e lo schermo attuale sarà memorizzato così com'e, senza alterare la finestra già definita.

Questo, tra l'altro, consente di avere a disposizione un vero e proprio editor grafico: si provi a pulire lo schermo con Shift + Clr/Home, poi, senza muovere la leva del joystick, si prema il fire.

Muovendo ora la leva, la finestra riapparirà sullo schermo vuoto, e, premendo ancora il pulsante di fuoco (non contemporaneamente al movimento), verrà "stampata" nella posizione in cui si trova in quel momento.

E' possibile, dunque, disegnare non solo con il cursore, ma anche usando la finestra come "pennello".

Infine è stata implementata anche la possibilità di escludere la visualizzazione del cursore premendo il tasto F1, mentre con il tasto F3 lo si riabilita.

Questo perchè, dato che la routine manipola l'interrupt di sistema, nella fase di trasferimento dello schermo, può talvolta rimanere impresso lo spazio in reverse del cursore.

In ogni caso, quando il "blink" (= lampeggio) non è visibile, è sempre possibile usare la tastiera, anche se alla cieca, e comunque è conveniente escludere il cursore solo quando si adopera il joystick per memorizzare lo schermo (il fire, insomma).

Non resta da aggiungere che la routine, pur abbastanza completa, è notevolmente migliorabile: si potrebbe, ad esempio, stampare la finestra anche durante il movimento, salvare il risultato su disco, e chi più ne ha più ne metta.

Difficile?

Niente affatto; anzi, per la verità, basterebbe davvero poco, ma questa è una soddisfazione riservata a chi ha già deciso di "immergersi" nella programmazione Assembly.

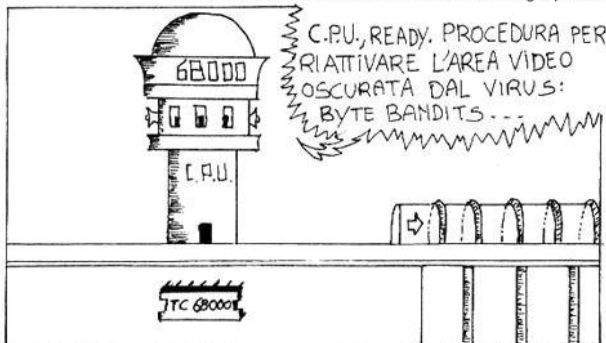
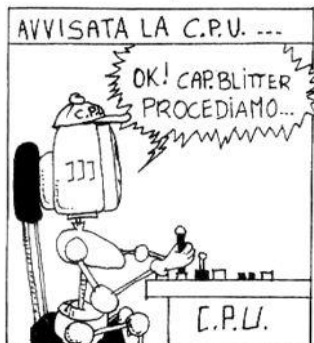
L'ALGORITMO GENERALE

E' il momento, dunque, di passare al disassemblato di queste pagine, edito con il solito Macro Assembler della Commodore.

Date le sue dimensioni non proprio... microscopiche, stavolta non ne esamineremo le singole istruzioni riga per riga, anche perchè, in massima parte, si tratta di tecniche e di codici mnemonici dell'Assembly già ampiamente descritti, nei precedenti inserti "Campus" (e non solo), dal n.52 in poi.

Il disassemblato, inoltre, è ampiamente commentato e diviso schematicamente nei vari moduli che lo compongono, per cui una sua lettura analitica dovrebbe risultare piuttosto facile.

Alcuni aspetti particolari della routine meritano tuttavia una certa attenzione, ed è di questi che ci occuperemo soprattutto. Prima è d'obbligo preci-

sare come il programma LM svolge il suo compito, ovvero il suo algoritmo generale.

Il movimento della finestra sullo schermo, in pratica, è realizzato mediante un semplice trasferimento di blocchi di memoria da un'area all'altra.

Chiariamo.

Quanto viene visualizzato sullo schermo in bassa risoluzione, è arcinoto, non è altro che il contenuto delle mille locazioni di memoria da $400 (1024) in poi.

Una finestra video, quindi, non sarà altro che un certo numero di locazioni all'interno della stessa area.

Ora, se noi utilizziamo altre due porzioni di memoria (= due buffer) e vi trasferiamo in una tutto lo schermo, e nell'altra il solo contenuto della finestra definita, per spostare quest'ultima in un'altra zona del video, basterà:

A) Copiare il contenuto del primo buffer, ovvero l'intera schermata, nella normale area video.

B) Trasferire il secondo buffer (= la copia della finestra) nelle locazioni desiderate dell'area video.

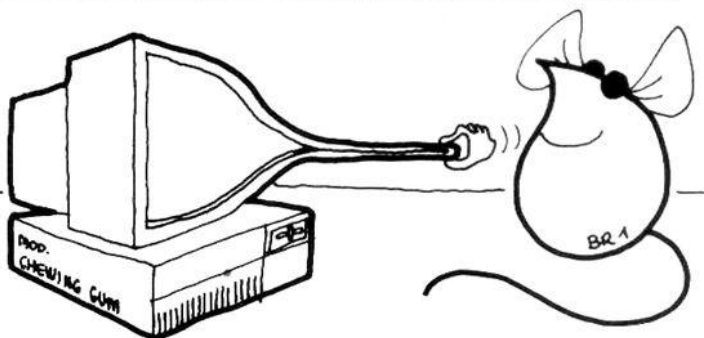
In tal modo, tra l'altro, si eviterà di alterare quanto era già visualizzato sullo schermo, finestra compresa.

Operando in LM, questi trasferimenti avvengono in termini di frazioni di secondo, per cui è possibile, affidando il controllo della variazione di posizione della finestra ad un joystick, ottenere un effetto di movimento vero e proprio, ed anche abbastanza veloce.

La nostra routine, oltre che a gestire quanto appena accennato, si occupa anche di parecchie altre cose, seguendo un flusso che può essere così schematizzato:

1) Preleva i parametri che delimitano la finestra (righe 22-30 del disassemblato), e controlla che non superino le dimensioni dello schermo (32-40).

2) Trasferisce l'area video in un buffer di eguali dimensioni, allocato da 50176 ($C400) in poi (righe 43-48).

3) Trasferisce in un secondo buffer (da 51200 in poi) l'area delimitata dai parametri scelti tramite la sys di attivazione (50-73).

LETTURA DEL JOYSTICK

PORTA 1 - REGISTRO $DC01 (dec.56321)

	binario	esad.	dec.
NON ATTIVA	1111 1111	FF	255
ALTO	1111 1110	FE	254
BASSO	1111 1101	FD	253
SINISTRA	1111 1011	FB	251
DESTRA	1111 0111	F7	247
FIRE	1110 1111	EF	239
ALTO+SINIS.	1111 1010	FA	250
ALTO+DESTRA	1111 0110	F6	246
BASSO+SINIS.	1111 1001	F9	249
BASSO+DESTRA	1111 0101	F5	245

PORTA 2 - REGISTRO $DC00 (dec.56320)

	binario	esad.	dec.
NON ATTIVA	0111 1111	7F	127
ALTO	0111 1110	7E	126
BASSO	0111 1101	7D	125
SINISTRA	0111 1011	7B	123
DESTRA	0111 0111	77	119
FIRE	0110 1111	6F	111
ALTO+SINIS.	0111 1010	7A	122
ALTO+DESTRA	0111 0110	76	118
BASSO+SINIS.	0111 1001	79	121
BASSO+DESTRA	0111 0101	75	117

Il processore 6510 offre un metodo estremamente veloce per memorizzare dati

4) Modifica l'interrupt di sistema in modo che, ogni sessantesimo di secondo, venga effettuato un controllo sulla eventuale attivazione del joystick, o pressione tasti F1 ed F3 (linee 75-93).

Se uno di questi eventi si è avverato, passa il controllo alle relative routine di gestione (94-168), altrimenti esce al normale proseguimento dell'interrupt, vale a dire $EA31.

QUALCHE DETTAGLIO... PER NIENTE SUPERFLUO

Se però si è premuto F1, l'uscita viene modificata in modo che punti a $EA61 (righe 86-87), impedendo così al sistema di provvedere alla visualizzazione del cursore.

Si tenga presente che tale procedura impedisce anche l'aggiornamento dell'orologio interno, per cui, qualora se ne dovesse far uso, si renderebbe necessario aggiungere un JSR $FFEA prima del salto ad $EA61.

Poco da dire sul punto 1, se non che utilizza una routine di sistema locata a partire da $E200, la quale, richiamata con JSR (=gosub), legge e riporta nel registro X un valore numerico compreso tra 0 e 255.

Qualora il numero non sia compreso in questo "range", viene rifiutato con un laconico Illegal Quantity Error.

La stessa segnalazione viene attivata, stavolta su controllo della nostra routine (riga 41), se viene definita una finestra che superi i limiti dello schermo.

Per la cronaca, basta un salto a $B248 (anche da basic: provate con sys 45640) per avere la suddetta segnalazione di errore.

Sempre a proposito della definizione della finestra, si noti (righe 34 e 39) che si comparano i parametri di riga e colonna con 27 e 42, e non 25 e 40, come ci si potrebbe aspettare.

Questo per supplire al fatto che, mentre noi "umani" preferiamo contare partendo da 1, il computer lo fa partendo da 0 (quindi: righe = 0/24, colonne = 0/39).

Con tale accorgimento è possibile inserire, dopo la sys, i valori nel modo che più ci è familiare.

Per quanto riguarda i trasferimenti della memoria da e verso i buffer, si è scelto di utilizzare due subroutine, una per lo schermo (175/191) ed una per la finestra (194/216), le quali sfruttano un indirizzamento indiretto postindicizzato (già trattato sui numeri scorsi) per svolgere il loro compito.

In tal modo, per dirigere il senso dello spostamento, è sufficiente settare opportunamente i soli puntatori in pagina zero.

Nella routine questi puntatori sono identificabili dalle label PRTS (= partenza schermo), ARRS (=arrivo schermo), PRTW (= partenza window) e ARRW (= arrivo window).

Quindi, per esempio, alle righe 43/48 vengono modificati i puntatori in modo che il punto di partenza sia l'area video, e quello di arrivo il buffer 1 (salto a Scrbuf).

Per il trasferimento opposto, i puntatori vengono invertiti (185/191) una volta per tutte, dato che tutti i trasferimenti successivi avverranno solo dal buffer 1 all'area video.

Lo stesso vale per la window, della quale, però, è necessario prima ricavare l'indirizzo effettivo dal quale comincia (52/63) secondo la formula:
inizio = 1024 +(X1 * 40).

UN'UTILE CATASTA

Sorge però un problema.

Nei trasferimenti dell'intera schermata i puntatori, pur essendo alterati nel corso della subroutine (vedi disassemblato), possono poi essere facilmente ripristinati, in quanto l'area video comincerà sempre da $400 e il buffer da $C400.

Per la finestra, invece, mentre l'indirizzo di buffer 2 è costante, il punto d'inizio sullo schermo viene modificato ad ogni spostamento.

Occorrerebbe, dunque, ad ogni trasferimento, ricalcolare il valore da inserire nei puntatori, o usare altre locazioni per memorizzarvi provvisoriamente il valore originario.

C'è però un altro metodo, molto più veloce, che consiste nel depositare l'indirizzo nello stack (vedi riquadro), per poi ricuperarlo dopo l'esecuzione della routine di trasferimento.

E' quello che potete rilevare alle righe 65/71 e 158/166, quasi identiche.

Già, ma perchè allora non le si sono incamerate in una subroutine unica?

Beh, solo per non complicare ulteriormente la routine.

Infatti, quando viene eseguito un JSR, il sistema deposita nello stack l'indirizzo di ritorno dalla subroutine, che quindi si "sovrapporrebbe" ai nostri valori (non si dimentichi mai la struttura LIFO).

Se il recupero dei valori che ci interessano avvenisse dunque nell'ambito di una subroutine, occorrerebbe prima prelevare e trasferire "al sicuro" i due byte che giovano al sistema, poi prelevare il nostro indirizzo, ed infine, prima di RTS, riporre i due byte nello stack.

Nulla di trascendentale, ma... meglio evitare complicazioni.

Prima di concludere, non resta che un breve ragguaglio sul controllo del joystick.

Le cosiddette "porte giochi" sono collegate ai registri del CIA $DC00 e $DC01.

Quando la leva del joy viene mossa in una delle quattro direzioni di base (des., sin., su, giù), viene azzerato un bit del nibble basso del relativo registro, secondo le modalità illustrate nella tabella di queste pagine.

Premendo il Fire, viene invece azzerato il bit 4.

Nella nostra routine (94/107), per leggere lo stato dei singoli bit, si fa scorrere verso destra (con LSR) il valore del registro $DC00, preventivamente depositato in accumulatore, in modo che il carry indichi se il bit è settato o meno.

E' questa, forse, la sezione che più si presta ad essere modificata.

Se in meglio o in peggio, dipende solo da voi.

UNO STACK PER DUE

Lo stack, vocabolo che sta per "pila" (intesa come catasta, niente a che vedere con walk-man e roba del genere), è una delle strutture basilari di qualunque sistema operativo, indipendentemente dal microprocessore adoperato.

Si tratta, semplicemente, di una particolare area di memoria, ad accesso molto veloce, in cui vengono memorizzate delle informazioni provvisorie, utili nel prosieguo di una routine.

Nel caso del nostro 6510 (e similari), la suddetta area è costituita dalla pagina uno, ovvero dalle locazioni da $100 ad $1FF.

La velocità di accesso a questa zona è data dal fatto che non è necessario precisare alcun indirizzo, in quanto la prima locazione libera dello stack è sempre puntata da un registro ad otto bit, interno al 6510: lo Stack Pointer (SP, o talora solo P).

In pratica, l'indirizzo della "sommità della pila" è sempre 1 nel suo byte alto, mentre il puntatore ne contiene il byte basso (quindi tra 0 e 255), e viene sempre aggiornato (automaticamente) dopo ogni operazione che riguarda la pila.

Importante da sottolineare è la particolare struttura dello stack, detta in breve LIFO (Last In First Out = Ultimo ad entrare, primo ad uscire), meglio spiegabile con un banale esempio.

Si immagini di compilare una serie di note scritte ognuna in un foglietto: man mano che vengono completate, si ripongono ordinatamente i fogli uno sopra l'altro.

Terminato il lavoro e volendo ricontrollare il tutto, il primo foglio che si prenderà in mano, quello in cima alla pila, sarà in effetti l'ultimo che si era scritto.

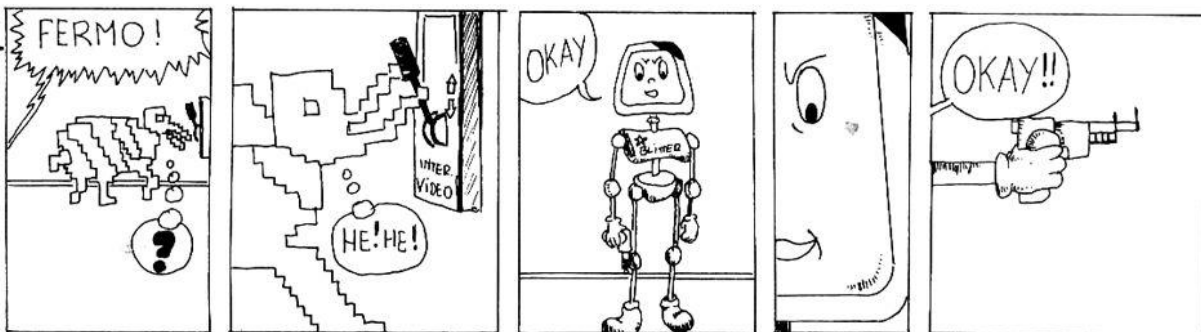
Tornando in argomento, occorre ancora ricordare che lo stack del 6510 procede, per così dire, in ordine inverso: il "fondo" della pila è la locazione $1FF, e, man mano che vi si deposita qualcosa, il puntatore viene decrementato.

Va da sè che l'ultima locazione disponibile, il "tetto", è la $100.

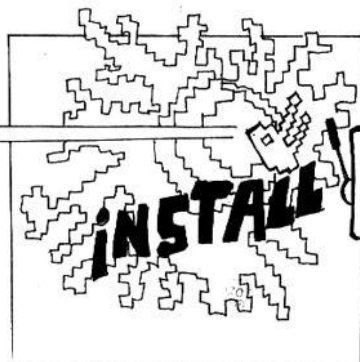
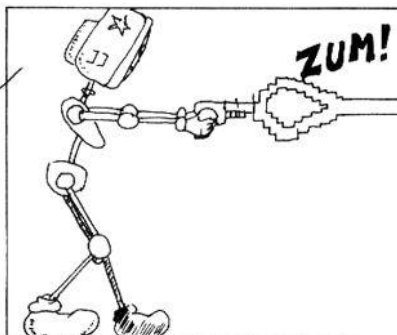
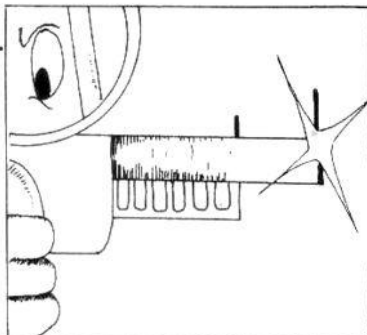
Il sistema adopera lo stack piuttosto intensamente, soprattutto per depositarvi gli indirizzi di ritorno da una subroutine, o per salvare il contenuto dei registri (A, X, Y e Status) prima di eseguire altre routine che li modificano.

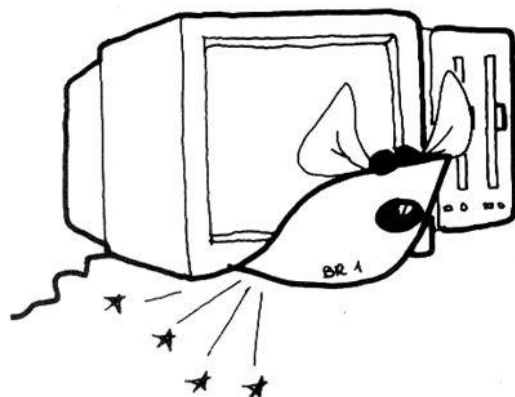
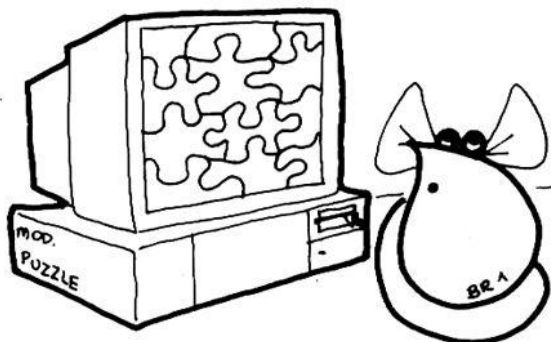
Tuttavia, consente anche al programmatore di accedervi, tramite le seguenti istruzioni dell'Assembly, obbligatoriamente riferite al registro Accumulatore, e a cui fa sempre seguito un aggiornamento dello Stack Pointer:

PHA ($48; PusH Accumulator on stack = spinge l'accumulatore) Deposita il contenuto di A alla sommità della pila.

PHP ($08; PusH Processor status) Deposita il contenuto del Registro di stato.

Usando lo stack nelle proprie routine, non bisogna mai dimenticare di lasciarlo così com'era prima del nostro intervento, per cui, in pratica, al numero di PHA presenti in un programma, deve SEMPRE corrispondere un egual numero di istruzioni PLA.

PLA ($68; PulL Accumulator = Tira fuori) Trascrive il contenuto della sommità della pila in accumulatore.
PLP ($28; Pone il contenuto della sommità della pila nel Registro di stato.

```
001 JOY     =$DC00       dec.56320
002 PRM     =$E200       dec.57856
003 SCR     =$0400       dec.1024
004 BF1     =$C400       dec.50176
005 BF2     =$C800       dec.51200
006 EXIT1   =$A7AE       dec.42926
007 EXIT2   =$EA31       dec.59953
008 ILLERR  =$B248       dec.45640
009 IRQ     =$314        dec.788
010 X1      =$F7         dec.247
011 Y1      =$F8         dec.248
012 X2      =$F9         dec.249
013 Y2      =$FA         dec.250
014 MAXRIG  =$02         dec.2
015 MAXCOL  =$A5         dec.165
016 PRTS    =$FB         dec.251
017 ARRS    =$FD         dec.253
018 PRTW    =$A7         dec.167
019 ARRW    =$A9         dec.169
020         *=$C000      Start=49152
021 ;----------------------------PARAMETRI---
022         JSR PRM
023         STX X1
024         JSR PRM      Preleva i dati
025         DEX          che seguono la
026         STX Y1       sys e li pone
027         JSR PRM      in 4 locazioni
028         STX X2       di pagina zero.
029         JSR PRM
030         STX Y2
031 ;-------------------CONTROLLO ERRORI
032         TXA          Somma il numero
033         ADC Y1       di righe e di
034         CMP #42      colonne della
035         BCS ERROR    finestra alle
036         STA MAXCOL   coordinate di
037         LDA X1       partenza. Esce
038         ADC X2       se il totale e'
039         CMP #27      superiore ai
040         BCC INIZIO   limiti dello
041 ERROR   JMP ILLERR   schermo.
042 ;-------------------SCHERMO IN BUFFER
043 INIZIO  STA MAXRIG
044         LDA #00      Setta puntatori
045         STA PRTS     di partenza ed
046         STA ARRS     arrivo e salta
047         STA ARRW     a subroutine di
048         JSR SCRBUF   trasferimento.
049 -------------------INIZIO FINESTRA-
050         LDA #>BF2
051         STA ARRW+1   Trova indirizzo
052         LDX X1       effettivo della
053         LDA #>SCR    locazione di
054         STA PRTW+1   inizio della
055         LDA Y1       finestra sullo
056 STORE   STA PRTW     schermo e ne
057         DEX          deposita il
058         BEQ CONT     valore nei rela-
059         CLC          tivi puntatori
060         ADC #40      di pagina zero.
061         BCC STORE
062         INC PRTW+1
063         BNE STORE
064 ;-------------------WINDOW IN BUFFER
065 CONT    PHA          Salva indirizzo
066         LDA PRTW+1   di inizio nello
067         PHA          stack e salta a
068         JSR TRASF2   trasferimento.
069         PLA          Preleva indir.
070         STA ARRW+1   dallo stack, lo
071         PLA          deposita nei
072         STA ARRW     punt. di arrivo
073         JSR WNDPNT   e setta partenza
074 ;-------------------MODIFICA IRQ--
075         SEI          Dirotta i pun-
076         LDA #<ROUT   tatori di IRQ
077         STA IRQ      verso riga 83
078         LDA #>ROUT   ed esce a rout.
```

```
079          STA IRQ+1     di esecuzione
080          CLI           prossimo comando
081          JMP EXIT1     basic.
082 ;------------------------TASTI F1 ED F3-
083 ROUT     LDA $CB       Se tasto <> F1,
084          CMP #4        salta,altrimenti
085          BNE CONFR2    dirotta uscita
086          LDX #$61      verso $EA61 (no
087          STX ESCE+1    cursor) ed esce.
088          BNE ESCE      Bne e' come JMP.
089 CONFR2   CMP #5        Se tasto = F3,
090          BNE CONFR3    ripristina nor-
091          LDX #$31      male uscita a
092          STX ESCE+1    IRQ di sistema
093          BNE ESCE      ($EA31) ed esce.
093 ;---------------------------JOYSTICK----
094 CONFR3   LDA JOY       Legge joy. ed
095          CMP #127      esce se non e'
096          BEQ ESCE      attivato.
097          LSR A         Legge i bit
098          BCC ALTO      che indicano
099          LSR A         la direzione
100          BCC BASSO     del movimento
101          LSR A         del joystick e
102          BCC SINIST    salta alle
103          LSR A         relative routine
104          BCC DESTRA    di gestione.
105          LSR A         Se Fire premuto,
106          BCS ESCE      trasferisce lo
107          JSR SCRBUF    schermo in buff.
108 ESCE JMP EXIT2         Uscita a IRQ.
109 ;---------------------------------------
110 ALTO     LDX MAXRIG
111          DEX           Controlla se
112          CPX X2        non si e' supe-
113          BEQ ESCE      rato il bordo
114          STX MAXRIG    superiore dello
115          SEC           schermo, e, se
116          LDA ARRW      no, sottrae 40
117          SBC #40       al puntatore di
118          BCS SALVA     arrivo della
119          DEC ARRW+1    finestra e salta
120          BNE SALVA     (BNE = JMP).
121 ;---------------------------------------
122 BASSO    LDX MAXRIG
123          INX           Come sopra, ma
124          CPX #27       controllando il
125          BEQ ESCE      bordo inferiore
126          STX MAXRIG    dello schermo
127          CLC           e sommando 40
128          LDA ARRW      (=1 riga) al
129          ADC #40       puntatore di
130          BCC SALVA     arrivo.
131          INC ARRW+1
132          BNE SALVA
133 ;---------------------------------------
134 SINIST   LDX MAXCOL
135          DEX
136          CPX Y2        Controllo ed
137          BEQ ESCE      incremento di
138          STX MAXCOL    una unita' del
139          SEC           punto di arrivo
140          LDA ARRW      della finestra
141          SBC #1        sullo schermo.
142          BCS SALVA
143          DEC ARRW+1
144          BNE SALVA
145 ;---------------------------------------
146 DESTRA   LDX MAXCOL
147          INX
148          CPX #42
149          BCS ESCE      Controllo e
150          STX MAXCOL    decremento dei
151          CLC           puntatori di
152          LDA ARRW      arrivo.
153          ADC #1
154          BCC SALVA
155          INC ARRW+1
156 ;-----------------------TRASFERIMENTI--
157 SALVA    STA ARRW      Salva puntatori
158          PHA           nello stack, poi
159          LDA ARRW+1    trasferisce lo
160          PHA           schermo e la fi-
161          JSR TRASF1    nestra dai buf.
162          JSR TRASF2    in area video.
163          PLA           Preleva valori
164          STA ARRW+1    originari dallo
165          PLA           stack, ripristi-
166          STA ARRW      na i puntatori
```

```
167          JSR WNDPNT   (anche partenza)
168          JMP ESCE     ed esce a IRQ.
169  ;-------------------TRASF. SCHERMO-
170  SCRBUF LDA #>SCR     Setta puntatori
171         STA PRTS+1    di trasferimento
172         LDA #>BF1     dallo schermo
173         STA ARRS+1    al buffer.
174  ;---------------------------------
175  TRASF1 LDX #04       Prepara indici
176         LDY #00       X ed Y.
177  LOAD1  LDA (PRTS),Y  Legge locazione
178         STA (ARRS),Y  e la deposita
179         INY           in buffer,
180         BNE LOAD1     incrementando
181         INC PRTS+1    i byte alti dei
182         INC ARRS+1    puntatori dopo
183         DEX           ogni "pagina"
184         BNE LOAD1     per 4 volte.
185         STX PRTS      Ripristina
186         STX ARRS      valori originari
187         LDA #>BF1     dei puntatori
188         LDX #>SCR     di partenza e
189         STA PRTS+1    di arrivo (prima
190         STX ARRS+1    low poi hi) e...
191         RTS           Return.
192  ;===================TRASF. FINESTRA=
194  TRASF2 LDX X2        X= righe window.
195  LOAD2  LDY #00       Y= colonna.
196  LOAD3  LDA (PRTW),Y  Legge locazione
197         STA (ARRW),Y  e la deposita.
198         INY           Continua fino a
199         CPY Y2        Y = colonne,
200         BNE LOAD3     dopodiche' si
201         DEX           incrementano
202         BEQ FINE      i puntatori
203         CLC           low, sia di
204         LDA PRTW      partenza che di
205         ADC #40       arrivo, di 40
206         STA PRTW      unita'.
207         BCC CONT2     Se la somma su-
208         INC PRTW+1    pera il valore
209  CONT2  CLC           255 (=cambio
210         LDA ARRW      di pagina), si
211         ADC #40       incrementano
212         STA ARRW      anche i byte
213         BCC LOAD2     alti degli
214         INC ARRW+1    stessi puntato-
215         BNE LOAD2     ri.
216  FINE   RTS           Return.
217  ;=================PUNTATORI WINDOW-
218  WNDPNT LDA #<BF2     Si settano i
219         STA PRTW      puntatori di
220         LDA #>BF2     partenza per
221         STA PRTW+1    trasferimento
222         RTS           buffer->schermo.
223  ;=================================
224         .END          ;Fine assemblaggio.
```

Beh, cari lettori, sembra che Giovedini, Iceman e Assembler abbiano apprezzato questo nuovo personaggio! Speriamo che sia così anche per voi ... Arrivederci!

SYS 64738

VUOI PUBBLICARE UN ANNUNCIO GRATUITO?

Commodore Computer Club ti offre la possibilità di pubblicare GRATUITAMENTE il tuo annuncio riguardo scambio o vendita di software, vendita o acquisto di apparecchi usati, ricerca di amici per fondare un club e così via. Utilizza lo spazio vuoto, incorniciato qui a fianco, usando un pennarello e tenendo presente che IL TUO STESSO ANNUNCIO verrà pubblicato così come ci perviene in Redazione, senza alcuna modifica. Ti consigliamo, pertanto, di usare una calligrafia chiara, a stampatello e dotata di caratteri sufficientemente grandi per consentire un'agevole lettura. Avranno precedenza gli annunci più curati graficamente e quelli di "carattere" marcatamente hobbystico. Invia l'annuncio in busta chiusa, affrancata secondo le norme vigenti, indirizzando a:

Commodore Computer Club
Viale Famagosta, 75
20142 MILANO

Hey! Vuoi diventare ricco? Vendo programmi "Professionali" per Totocalcio per C64, Amiga, IBM. OK!!! Maurizio Ascione, Via Panoramica, 10 - 80056 Ercolano (Na). Tel. 081/7392240 (Ore 16.00).

Commodore Club Foggia vende programmi per CBM64 a bassissimo costo. Abbiamo tutte le ultime novità direttamente dall'Inghilterra. Per informazioni: Armando: Tel. 0881/35484 (ore pasti) - Giuseppe: 0881/35255 (ore pasti).

Vendo per C64 "Target Renegade", "Giana Sisters", "Seuck", "Dubble Bobble", "Ritorno dello Jedi", "Pac Land", "Cybernoid 2", "Flinstones", "Netherworld" e altri a L. 1.000 (Min. 20 su cassetta) o scambio con "Zzap!" e/o C.C.C. arretrati. Ciciotti Mario, Via Vidimari 41 - 67051 Avezzano (Aq).

Vendo programma "The 64 Emulator" per Commodore Amiga + istruzioni in italiano a L. 20.000 (c.F.r. C.C.C. n. 60). Telefonare ore serali allo 004191/435356. Vendo e scambio PRG X Amiga 500!!!

La Biotech Simbion vi offre la possibilità di acquistare programmi di ogni livello (Games, utilitis, gestionali) per Amiga a L. 4.000 più disco. Per informazioni telefonare allo 02/514410 dalle ore 19.00 in poi. Siamo fortissimi!

Scambio programmi per C128, sia in modo 128 che in CP/M, vasta biblioteca software con oltre 350 titoli. Inviare liste a: Albanese Rosario, C.so Alberto Amedeo, 66 - 90138 Palermo. Tel. 091/327764

Vendo C64 + 2 registratori + 2 Joystick + 2 Cartridges + 31 cassette con 600 giochi e utilities + manuali e istruzioni. Tutto in ottime condizioni a L. 300.000. Stefano: tel. 0586/402670

Vendo per CBM64 giochi d'edicola e originali come Last Ninsa 2, Double Dragon, Salamander e tanti altri sia su disco che cassetta. Epifanio Fabio, Via Spazzapan 8 - 10135 Torino. Tel. 011/614648

C64 + reg. + Drive 1541 + MPS 803 + adattatore telematico + mouse + 300 dischi (6000 prog.) + porta dischi L. 1.350.000. Rizza Vincenzo, Via Garibaldi 37 - 21012 Cassano Magnago (Va). Tel. 0331/280292 (sera).

Vendo C128 + Disk Drive 1571 + Joystick + giochi + programmi a L. 600.000. Telefonare o scrivere a Francesco Trainiti, Via Sistiana n. 32 - 33100 Udine. Tel. 0432/503146

Causa passaggio a sistema superiore, svendo software X C64 e C128 a prezzi irrisori, anche ultime novità. Richiedere lista e informazioni gratuite. A residenti zone vicine ad Adria (Ro). Vendo il mio sistema completo. Contattatemi. Bisco Gabriele, Via P. Nenni 13 - 45011 Adria (Ro).

Vendo C128 Personal Computer come nuovo (con manuali ed imballo) + Drive 1541 + registratore originale Commodore + migliaia di programmi disco e cassetta (Utility-Games) + (s.o. CP/M Plus V3.0-Oxford Pascal) + Geos a L. 790.000. Andrea Lucarelli, Via dei Giordani - 00199 Roma. Tel. 06/8393226

Vendo cassette da 46 min. con giochi e utility recentissime da L. 7.000 e dischi da L. 3.500 l'uno. Per maggiori informazioni o per richiedere liste telefonare allo 080/917865 ore 15.00 e chiedere di Leo.

Vendo C128 (1 anno), D.D. 1541, stampante MPS 803, Joystick prof., disco CPM, copritastiera, registratore, centinaia di giochi e programmi, libri, riviste, compilatori LM e Pascal, a L. 700.000. Vanni Boncinelli - 50135 Firenze. Tel. 055/609556

Vendo C64 + registratore + copritastiera + 50 cassette (con moltissimi programmi e relativa documentazione) + manuali basic ed assembler. Imballi originali, tutto come nuovo. L. 270.000 trattabili. Montanari Pierluigi. Telefonare ore pasti allo 0331/930311

Causa passaggio a sistemi più evoluti vendo C128D + 400 programmi e giochi a livello di Dobble, Cal. Games, R-Type ecc. tutto a L. 700.000. Tel. 0746/485287. Via Consoni 14 - Rieti.

Cerco e offro Software X C64, nastro e disco. Chiamare Michele (ore pasti): tel. 3966764. Solo Roma.

Vendiamo n.s. Software originale (da noi prodotto) a bassissimi prezzi (solo zona Roma). Rivolgersi allo 3060407 - 3966764 ore pasti. Si garantisce massima serietà!

Ti interessa scambiare o comprare (L. 2.000 cad.) ultime novità in fatto di giochi? Bene, se hai un C64 e un Drive, sto cercando proprio te altrimenti amici in tutta Italia (preferibilmente in zona). Alessandro Zarilli, Via delle Magnolie 5 - 53036 Poggibonsi (Si). Tel. 0577/936152

Vendo Drive C=1541 II + (in regalo!) "Super Fast Load" ottimo stato, 1 anno di vita. Raffaele: tel. 02/733610

Cercasi disperatamente ultime novità programmi e nuovissimi giochi per C64 e C128. Rivolgersi a Flavio: tel. 06/3378638. Cap 00168 Roma. Chiedo ad Alessandro Bordon la città dove abita (nome).

Vendo C128, registratore, drive 1570, 91 dischi un'enciclopedia completa sul 64 e V20 a lire 600.000. Nicola Giomo, Via Liguria n. 8 - 30026 Porto Gruaro (Ve). Tel. 0421/72941

Vendo 128/D a L. 600.000, Corol 128 L. 60.000, "Grande guida 128 per il programmatore" ed altri testi a L. 50.000. Per informazioni: tel. 051/515565. Maurizio Romboli, Via Salgari 47 - 40127 Bologna.

Vendo programmi C64 , Amiga ultime novità arrivi settimanali. Abbonamenti a prezzi modici per informazioni telefonare a Massimo: 0322/65169 (ore pasti).

Vendo floppy 5 1/4-3 1/2 DS DD a prezzi eccez. Scambio software per MS-DOS compatibili. Cerco amici in zona Chieri (To) per formare un club. Telefonare: 011/9478960

Vendo CBM 128 + Disk Drive + registratore + (monitor a colori) + 2 Joystik + 2 portadischi + 175 dischi, 80 cassette + Paddles + Niki II Cartrige. Prezzo trattabile. Chiedere di Marcello. Telefonare ore pasti all'0575/20251

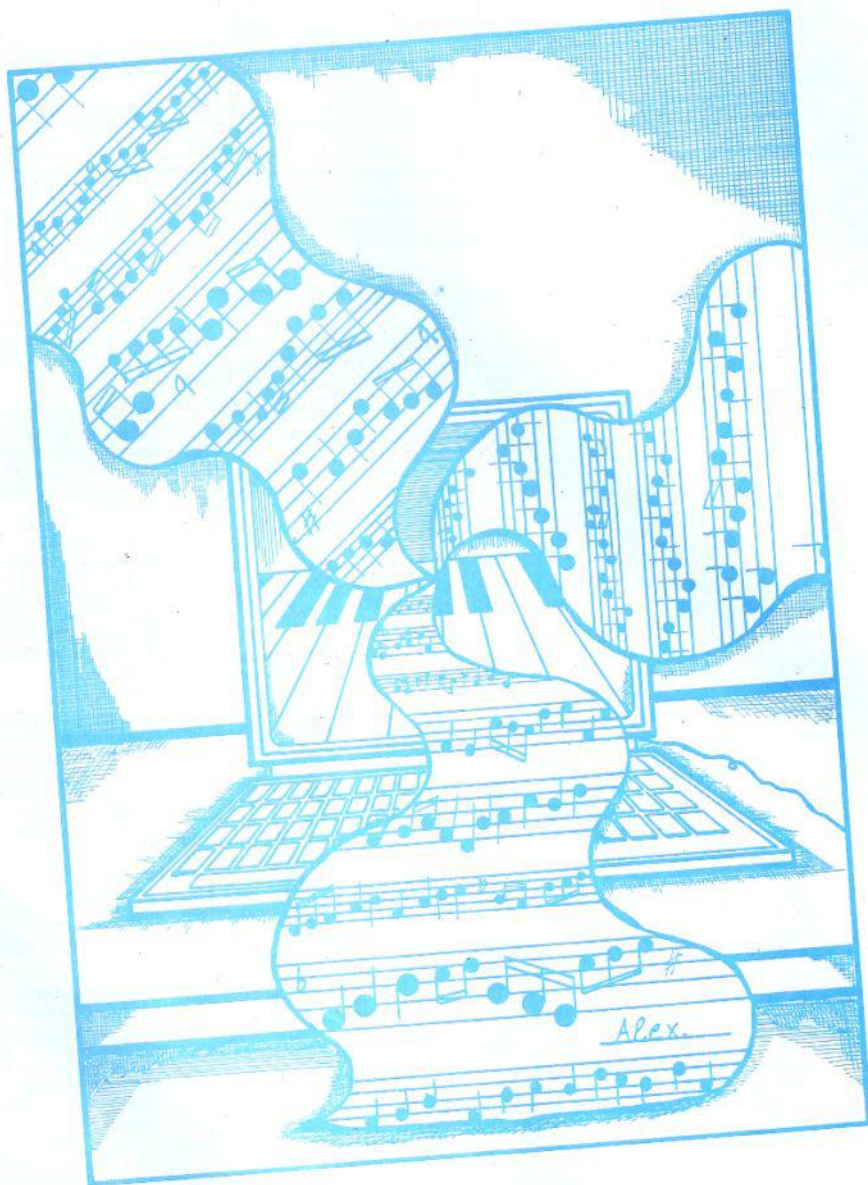
Amiga/C64/IBM-Manuali/Software-Nutrite liste. Arrivi settimanali. Ultime novità inglesi. Supersconti agli abbonati. Mailsoft Group by Diego: Via Monte Amiata 16 - 02015 Cittaducale (Rieti). Tel. 0746/62728

Vendo a prezzo contenuto (molto contenuto) Commodore VIC20 + registratore (in ottime condizioni). Per informazioni telefonare all'051/822230. Fiorini A.N.D.

A tutti i possessori del C128 ed adattatore 6499: ma funziona quest'ultimo sul 128? Si potrà mai sapere la verità sulla compatibilità. Fatemi sapere la vostra esperienza su questa rivista. Solla Federico Tel. 442428 (081) - Napoli.

La Polo Soft ha avviato una Campagna abbonamenti Software per C64 - c/o Luca Pelusi - via P. Picasso, 4 - 64023 Mosciano S. Angelo (Te) - Tel. 085/866280

Amiga: appassionato di Animazione cerca entusiasti utilizzatori di Modeler/V. 3D x creativo rapporto di collaborazione. Amigo Simone - tel. 0577/222867 (ore pasti).

Scambio software per C128 ricambio con programmi per C128 e ultime novità per C64. Inviare liste a: Scarpa Francesco - via Prota n. 69 Torre annunziata (NA) - cap 80058 - tel. (081) 5361553.

COME PAVAROTTI CON IL C/64

Una semplice teoria musicale, ed un utile strumento software, possono costringere un motivo musicale ad adeguarsi alla vostra voce

di **Valentino Spataro**

Questo pubblicato non è il solito programma per sentire melodie più o meno affasci
nanti; non è un ennesimo complicato metodo per suonare canzoni con il computer; neppure una rassegna di effetti speciali o nuovi suoni: qui gli unici a far musica siete voi. E allora?

Pensate di essere in estate, a pochi metri da un bel mare pulito(!), con la Luna che illumina il vostro spartito di Baglioni, la chitarra ed il consueto grappolo di belle ragazze perdutamente innamorate di voi.

Ma ecco che arriva quella nota un po' più alta delle altre, che solo Baglioni sa fare, e voi o fate una stecca o scendete di una ottava rovinando l'atmosfera.

Certo che se si potesse risolvere il problema non sarebbe male. MA ORA SI PUO'! Basta abbassare tutta la canzone di qualche semitono ed il gioco è fatto. Il programma provvede a questo e a molti altri piccoli problemi che assillano tutti noi che ci divertiamo a suonare la chitarra o il clarinetto o altri strumenti d'accopagnamento sulle spiaggie d'Italia.

A CHE SERVE IL PROGRAMMA

Il programma permette di stampare i soli accordi (in modo da avere sempre con voi pochi fogli con gli accordi delle canzoni più belle), o i soli testi (da dare ai vostri amici mentre suonate), o entrambi (per suonare e insieme cantare la canzone). Oltre a questo permette notevoli facilitazioni in fase di immissione dei dati, avvalendosi della funzione "copia accompagnamento" (dato che generalmente l'accompagnamento delle varie strofe è sempre lo stesso, risparmiando così un bel po' di tempo nella fase di immissione dei dati). La canzone, inoltre, viene memorizzata in formato leggibile da EASY SCRIPT, per poi personalizzarla ulteriormente (ma se la modificate ricordate di non cancellare il file di partenza perchè è l'unico leggibile da programma); è possibile inoltre alzare o abbassare l'accompagnamento di una strofa di X semitoni (per chi non lo sapesse 1 tono = 2 semitoni) con un semplicissimo comando; o cambiare la notazione di una strofa da inglese a italiana con altrettanta semplcità per chi non conosca una delle due notazioni.

Uno strumento utilissimo, quindi, per le più diverse esigenze.

COME USARE IL PROGRAMMA

Descriviamo ora le singole opzioni del programma:

UN ESEMPIO PRATICO

Un conto è sapere a cosa serve una opzione, un altro usare al meglio il programma. Proviamo quindi a inserire insieme una canzone con il programma.

Prendiamo una canzone famosa: "L'anno che verrà" di Dalla (alcune strofe sono presenti anche su "Directory" di questo mese).

Innanzitutto indichiamo in quale notazione è scritta la canzone: da menù principale (che d'ora in poi chiameremo solo menù) premiamo "N" e poi 2 per indicare la notazione italiana. Premiamo ora "E": pulito lo schermo comparirà il numero di strofa in cui stiamo introducendo i dati. Alla richiesta del testo inseriamo le prime parole della canzone ("Caro amico ti scrivo") e, alla richiesta di AC-C. (accompagnamento), in corrispondenza delle singole parole (spostandosi con il cursore per posizionarsi correttamente) anche l'indicazione della tonalità degli accordi (DO+, SOL8-7+, FA#8, ecc.).

Confermati gli inserimenti, vedremo in alto la richiesta della successiva frase da inserire. Se alla richiesta del testo digitiamo, come primo carattere, un più (+) o un meno (-) protremo scorrere le frasi dell'attuale strofa e, eventualmente, modificarle (+ e - non verranno memorizzati nel testo). Invece scrivendo, alla richiesta del testo, la sola lettera "A" o "T" potremo vedere l'intera strofa, solo accompagnamento o solo testo.

Proseguiamo a immettere le frasi della prima strofa avvalendoci, se necessario, di questi aiuti. Terminata l'operazione scriviamo "F" alla richiesta del testo; alla richiesta di memorizzare la strofa rispondete "S". Se rispondete "N", per errore, non abbiate paura perchè le strofe restano in memoria anche se non sono considerate della canzone. Scegliendo nuovamento Edit, da menù, e scrivendo alla richiesta del testo la sola lettera "F", e confermando, recupererete la strofa.

Passiamo ora alla seconda strofa. Essa ha gli stessi accordi della prima; possiamo quindi copiarli in questa seconda strofa premendo da menù "G" (gestione) e vedendo la prima strofa premere "C", scegliere di copiare solo l'accompagnamento e indicare alla richiesta la strofa n. 2 (non ancora esistente, ma che viene subito creata). Restando nella gestione strofe, visualizzare, con "T", la strofa 2, che sarà vuota di testo ma che troverete piena degli accordi premendo A; con l'opzione M (modifica) aggiungere il testo della strofa 2. Un'altra strada è quella di scegliere da menù "E" e inserire solo il testo della seconda strofa, memorizzare e poi premere da menù "G", visualizzare la prima strofa e chiedere con "C" di copiare solo l'accompagnamento nella strofa n. 2. In quest'ultimo caso ricordarsi di controllare che gli accompagnamenti siano in giusta corrispondenza con le parole, magari effettuando una stampa.

A volte capita che una strofa differisca da un'altra solamente per la tonalità: basterà copiarla come spiegato sopra e poi, visualizzandola tramite i tasti + e -, premere S e indicare di quanti semitoni deve essere innalzata o abbassata: se la si deve abbassare di 3 semitoni basterà scrivere -3 quando richiesto.

Quando siete nella gestione delle strofe ricordatevi che se premete N, e chiedete di cambiare la notazione, la strofa verrà inserita nella notazione opposta a quella indicata prima della conferma, e non controlla se la strofa è stata già trascritta nella nuova notazione. Per cambiare notazione, o spostare di tonalità una intera canzone, bisogna agire sulle singole strofe ripetendo le operazioni di volta in volta. Nel caso sia stata cambiata la notazione di un'intera canzone, ricordatevi di cambiare, da menù, la notazione della canzone.

Terminata la digitazione di una canzone ricordatevi di registrarla, e assicuratevi che compaia il messaggio "saving strofa x/y" dove "x" è la strofa in corso di registrazione ed "y" il numero totale di strofe. Ricordatevi inoltre che, come in tutti i computer MS-DOS, se registrate con il nome di un file già esistente, cancellerete automaticamente la vecchia versione.

Usando il programma noterete che il numero massimo di strofe memorizzabili è 10. Basterà comunque modificare tale valore inserendo quello desiderato nella variabile MS all'inizio del programma (Es. MS=20). Per i più bravi: non si consiglia di modificare con altrettanta semplicità il valore di MF, se non dopo aver considerato la possibilità di caricare canzoni le cui strofe hanno MF diversi.

Con un po' di allenamento vi accorgerete che il programma risulterà utilissimo per suonare canzoni che, nella versione originale, risultano "troppo" alte per le vostre possibilità canore; oppure per testi scritti nella notazione anglosassone. Oltre alla possibilità di scambiarvi, tra amici, canzoni leggibili da EASY SCRIPT.

N NOTAZIONE

Innanzitutto, prima di introdurre una canzone (opzione E = edit da menu) o dopo averne caricata una (L = load da menu) o dopo aver cambiato la notazione a tutte le strofe (N = notazione da G = gestione strofe) bisogna indicare al computer in quale notazione è scritto l'accompagnamento della canzone con questa opzione (diversa da quella citata prima e che descriveremo in G = gestione).

SE LA NOTAZIONE INDICATA NON E' QUELLA USATA NELLA CANZONE, LE FUNZIONI DI EDITAZIONE E GESTIONE DELLA CANZONE NON FUNZIONERANNO CORRETTAMENTE.

E EDIT STROFE

Serve a introdurre i dati in memoria. Verranno richiesti, di ogni singola strofa, testo e accompagnamento, da inserire come nel formato indicato nel riquadro. La scala dell'accompagnamento va indicata esattamente come la trovate nelle linee DATA da 1570 a 1610 e cioè: le scale maggiori esigono, dopo l'indicazione della tonalità, il segno positivo (+); quelle minori il segno negativo (-); inoltre il simbolo di cancelletto (#) indica i diesis e quello di percentuale (&) i bemolli. Dopo questa indicazione potrete scrivere tutte le altre informazioni della canzone. Facciamo degli esempi: è corretto scrivere:

DO+7 SOL&+ FA#-6 SI- dim

NON è corretto invece

DO7 SOL bem+ FA#6- SI dim-

Se, inoltre, alla domanda del testo scrivete "A" oppure "T" potrete vedere, della strofa che state immettendo, rispettivamente l'accompagnamento o il testo. Premendo "+" e "-" potete passare alla frase successiva o precedente per modificarla. Quando avete finito premete "F" e il programma chiederà se la volete memorizzare. Se rispondete "S" la memorizzerà, se "N" potrete modificare i dati prima immessi, se "E" tornate al menù senza memorizzare. Se, però, scegliete di nuovo questa opzione da menù troverete ancora i dati precedentemente immessi che potrete modificare a piacere.

G GESTIONE STROFE

L'opzione permette di "listare" la canzone (premendo "A" listerà solo l'accompagnamento, "T" solo il testo) e potrete scorrere le strofe premendo "+" oppure "-". Premendo "N", e confermando la scelta, potrete cambiare la notazione della strofa che state vedendo; premendo "S" potrete spostare la canzone di X semitoni (-12 < X < 12). Queste due ultime opzioni permettono di vedere come il programma provvede alle sostituzioni: può essere utile per chi abbia la buona volontà di imparare a farlo da solo quando non abbia la disponibilità del computer. Premendo il tasto "M" si può modificare una frase (testo e accompagnamento) della strofa che vedete. "C" copia il testo o l'accompagnamento, o entrambi, dell'attuale strofa in un'altra. Se in memoria ci sono 5 strofe, e chiedete di copiarla nella sesta, creerete una nuova strofa. Torna utile quando si deve ripetere lo

BREVE GLOSSARIO

Sia nell'articolo, che nel programma, sono presenti alcuni termini che sono stati usati sempre con lo stesso significato ma talvolta con una accezione ben precisa. E' bene quindi dare un breve sguardo a questi significati.

Notazione:

Serve a indicare quella convenzione secondo la quale sono scritte le note. La nota "LA" è così nella notazione italiana, ma in quella inglese è "A"

Strofa:

E' l'unità fondamentale di una canzone. Può essere una strofa vera e propria O ANCHE un ritornello.

Frase:

Le strofe sono divise in frasi. Ogni frase è composta dal testo e dal relativo accompagnamento

Formato:

E' il formato in cui si devono scrivere le frasi con il programma proposto. E' bene spendere qualche parola in più.

I testi di musica classica hanno il pentagramma con le note da suonare per le due mani. I testi di operetta hanno vari pentagrammi: per esempio due per la parte per pianoforte ed una per il cantante; sopra il pentagramma, inoltre, saranno presenti le parole che andranno cantate secondo le indicazioni del pentagramma.

Gli spartiti di musica leggera per pianoforte sono fatti generalmente così. A volte, però, manca del tutto la parte per pianoforte ed è presente solo il pentagramma con la melodia (cioè la parte del cantante) e alcune indicazioni di scale (DO, FA# min, ecc.). A volte manca del tutto il pentagramma e trovate solo il testo e le scale. Suonare questi spartiti non è difficile: è solo diverso. Tenete presente che in questo ultimo formato la melodia non è segnata ma la fate voi cantando. Alle vostre mani resta il compito di inventare l'accompagnamento come più preferite.

Concludendo, il programma memorizza il testo ed il corrispondente accompagnamento secondo questo formato:

RE (Caro amico ti sc-)
SOL (rivo, co-)
LA7 (sì...)

Con un po' di allenamento vedrete che non è assolutamente difficile ed anzi è molto più divertente perchè siete voi a modificare la canzone secondo la vostra sensibilità.

stesso accompagnamento per strofe diverse o per i ritornelli. "F" torna al menu.

P STAMPA TESTO

Una volta indicato il nome della canzone, e di quale strofa volete la stampa, il programma chiederà se volete stampare il testo, l'accompagnamento o entrambi, e procederà.

L LOAD
e S SAVE

Caricano e salvano su disco le canzoni. Se usate la versione 2 del GWBASIC fate attenzione perchè il comando save cancella il vecchio file su disco per registrare quello nuovo SENZA AVVERTIRE, come, del resto, avviene su tutti i computer MS-DOS. Per chi usa la versione 1 del GWBASIC si devono apportare le seguenti modifiche:

```
570 open 2, 8, 2, space$ (5) + f$ + "cnz,
s, r": open 1, 8, 15
580 input#1, a, a$: print a$: close 1: if
a<>0 then close #2: goto 550
470 open 2, 8, 2, space$ (5) + f$ + "cnz,
s, w": open 1, 8, 15
480 input#1, a, a$: print a$: close 1: if
a<>0 then close #2: goto 450
```

I file prodotti dalle due versioni sono compatibili tra di loro.

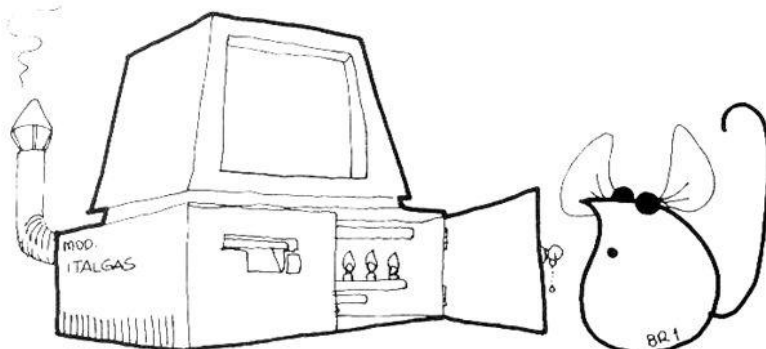
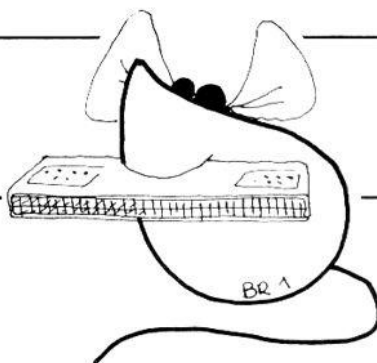
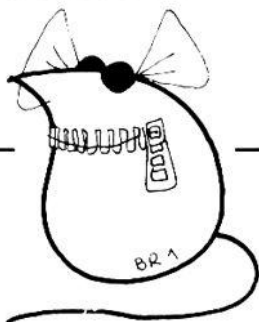
LE VARIABILI DEL PROGRAMMA

Questa è la parte che si può non leggere (sempre ammettendo che siate riusciti a star dietro al resto) ma per chi vuole divertirsi di più è la parte più interessante.

Brevemente:

MS = numero massimo di strofe
MF = numero massimo di frasi per strofa
PA$ (MS, MF) = contiene il testo
AC$ (MS, MF) = contiene l'accompagnamento
NO$ (25, N) = contiene le scale (V. n DOPO)
N = 0 notazione italiana
N = 1 notazione inglese
M$ = "T" fa vedere solo testo
M$ = "N" fa vedere solo accompagnamento
S = indica quale strofa è l'attuale
FL$ = nome del file

```
90 REM EDITOR DI CANZONI BY SPATARO VALENTINO
91 REM PROGRAMMA PER C/64 DOTATO DI EMULATORE G-BASIC
92 REM OPPURE PER COMPUTER MS-DOS COMPATIBILI DOTATI DI GW-BASIC
100 :
105 ON ERROR GOTO 1620
110 MS-10:REM MS-MASSIMO NUMERO DI STROFE
120 MF-20:REM MF-MASSIMO NUMERO DI FRASI PER STROFA
130 DIM NO$(25,1),PA$(MS,MF),AC$(MS,MF):N-1:REM N-1 NOTAZIONE ITALIANA
140 M$-"T":REM VISUAL TESTO
150 FORA-1TO24:READ NO$(A,0),NO$(A,1):NEXT
160 REM MENU' PRINCIPALE
170 CLS:PRINT"MUSICA & COMPUTER":PRINT
180 PRINT "N NOTAZIONE ";:IF N-0 THEN PRINT"INGLESE":ELSE PRINT"ITALIANA
190 PRINTTAB(6)"STROFE:":PRINT"E EDIT":PRINT"G GESTIONE
200 PRINTTAB(6)"CANZONE:":PRINT"P STAMPA"
210 :
220 PRINT"S SAVE":PRINT"L LOAD":PRINT"F FINE
230 PRINT"LA CANZONE '"FL$"' HA"WS"STROFE"
240 GET A$:IFA$-"" GOTO240
250 ON INSTR("FGSLEPN",A$) GOTO 270,280,440,540,660,810,1280
260 BEEP:GOTO240
270 END
280 REM GESTIONE STROFE
290 S-1:F-1
300 CLS:GOSUB 1320
310 LOCATE 22,1:PRINT"M-MODIFICA FRASE  +/- STROFA N."S"/"WS
320 PRINT"C-COPIA  S-SPOSTA TONALITA'"
330 PRINT"T-TESTO  A-ACC. F-FINE N-NOTAZIONE"
340 REM PRINT"+/- ALTRA STROFA. ATTUALE N."S;
350 GET A$:IFA$-""GOTO350
360 ON INSTR("FMI+-TASCN",A$) GOTO 160,380,440,400,410,420,430,900,1190,1630
370 BEEP:GOTO350
380 GOSUB 1530:F-0:INPUT"INSERIRE N.FRASE DA EDITARE ";F:IF F-0 GOTO310
390 GOSUB 1420:GOTO300
400 S-S+1:IF S>WS THEN S-1:GOTO 300:ELSE GOTO 300
410 S-S-1:IF S<1  THEN S-WS:GOTO 300:ELSE GOTO 300
420 M$-"T":GOTO300
430 M$-"N":GOTO300
440 REM ROUTINE DI SAVE
450 F$-"":INPUT "SAVE CANZONE (D)";F$:IFF$-"D" THEN CLS:FILES:GOTO450
460 IF F$-"" GOTO160
470 OPEN F$+".CNZ" FOR OUTPUT AS #2
480 IF ER<>0THEN ER-0:PRINT ERDEV$:CLOSE #2:GOTO450
490 FOR A-1 TO WS:PRINT"SAVING STROFA"A"/"WS:FOR B-1 TO MF
500 A1$-PA$(A,B):A2$-AC$(A,B):IF A1$-"" THEN A1$-"*"
510 IF A2$-"" THEN A2$-"*"
520 PRINT#2,A2$:PRINT#2,A1$
530 NEXT:NEXT:FL$-F$:CLOSE #2:GOTO160
540 REM ROUTINE DI LOAD
550 F$-"":INPUT "LOAD CANZONE (D)";F$:IFF$-"D" THEN CLS:FILES:GOTO550
560 IF F$-"" GOTO160
570 OPEN F$+".CNZ" FOR INPUT AS #2
580 IF ER<>0THEN ER-0:PRINT ERDEV$:CLOSE #2:GOTO550
590 A-0
600 A-A+1:PRINT"LOADING STROFA"A:FOR B-1 TO MF
610 INPUT#2,A2$,A1$
620 IF A2$-"*" THEN A2$-""
630 IF A1$-"*" THEN A1$-""
640 PA$(A,B)-A1$:AC$(A,B)-A2$:NEXT:IF EOF(2)-0 GOTO600
650 FL$-F$:WS-A:CLOSE #2:GOTO160
```

```
660 REM ***  EDIT STROFE  ***
670 S=WS+1:F=1:IF WS=>MS THENPRINT"MEMORIA PIENA":GOSUB 1400:GOTO160
680 IF F>=MF GOTO 790:ELSE CLS:PRINT"STROFA N."S"FRASE"F
690 PRINT"SCRIVI +/-/T=TESTO/A=ACC./F=FINE IN TESTO"
700 LOCATE 5,1:PRINT "TESTO "PA$(S,F):LOCATE 5,1:LINEINPUT "TESTO ":A1$
710 IF A1$="T" THEN M$="T":CLS:GOSUB 1320:GOSUB 1400:GOTO680
720 IF A1$="A" THEN M$="N":CLS:GOSUB 1320:GOSUB 1400:GOTO680
730 IF LEFT$(A1$,1)="+" AND F<MF+1 THEN F=F+1:GOTO 680
740 IF LEFT$(A1$,1)="-"AND F>1 THEN F=F-1:GOTO 680
750 IF A1$="F" GOTO 790
760 LOCATE 10,1:PRINT "ACC.  "AC$(S,F):LOCATE 10,1:LINEINPUT "ACC.  ";A2$
770 GOSUB 1500:IF A$="N" GOTO 700
780 PA$(S,F)=A1$:AC$(S,F)=A2$:F=F+1:GOTO 680
790 PRINT "VUOI MEMORIZZARE ?":GOSUB 1510:IF A$<>"S" GOTO 160
800 WS=WS+1:GOTO 160
810 CLS:INPUT"NOME CANZONE ";Q$:PRINT"IN MEMORIA "WS"STROFE"
820 SA=0:INPUT "STAMPO DA STROFA ";SA:IF SA=0 GOTO160
830 EA=0:INPUT "STAMPO A STROFA ";EA:IF EA=0 OR SA>EA OR EA>WS GOTO160
840 GOSUB 1360:IF M1$="F" GOTO 160
850 LPRINT "NOME CANZONE '"Q$"' DA"SA"A"EA"SU"WS
860 FOR A=1 TO WS:LPRINT:FORB=1 TO MF:A1$=PA$(A,B):A2$=AC$(A,B)
870 IF (M1$="2" OR M1$="3") AND A2$<>"" THEN LPRINT A2$
880 IF (M1$="1" OR M1$="3") AND A1$<>"" THEN LPRINT A1$
890 NEXT:LPRINT:NEXT:GOTO 160
900 REM SPOSTA DI X SEMITONI L'ACCOMPAGNAMENTO
910 :
920 GOSUB 1530:Q=0:INPUT "SPOSTO DI QUANTI SEMITONI ";Q:IF Q=0 GOTO 300
930 PRINT"WAIT";
940 IF ABS(Q)>11 GOTO920:REM NON HA SENSO >12 SEMITONI PERCHE' >1 OTTAVA
950 CLS:M$="N":GOSUB 1320:FOR A=1 TO MF:A$=AC$(S,A):IFA$="" GOTO 1170
960 FOR V=1TO LEN(A$):FOR B=1 TO 24
970 IF MID$(A$,V,3)="   " THEN V=V+2:GOTO1160:REM 3 SPAZI
980 IF MID$(A$,V,1)=" " GOTO1160
990 IF F$="N"THEN AX$=NO$(B,N2):AB$=NO$(B,N1):GOTO 1010
1000 AB$=NO$(B,N)
1010 IF MID$(A$,V,LEN(AB$))<>AB$ GOTO1160
1020 IF F$="N" GOTO1080
1030 X=B+Q:IF B<13 AND X>12 THEN X=X-12
1040 IF B<13 AND X<1  THEN X=X+12
1050 IF B>12 AND X>24 THEN X=X-12
1060 IF B>12 AND X<13 THEN X=X+12
1070 AX$=NO$(X,N)
1080 XL=LEN(AX$):BL=LEN(AB$):A$=LEFT$(A$,V-1)+AX$+RIGHT$(A$,LEN(A$)+1-V-BL)
```

```
1090 IF XL=BL GOTO 1150
1100 IF XL>BL GOTO 1110:ELSE GOTO 1130
1110 Z=INSTR(V,A$," "):IF Z=0 GOTO 1150
1120 A$=LEFT$(A$,Z)+RIGHT$(A$,LEN(A$)-Z-XL+BL):GOTO 1150
1130 Z=INSTR(V,A$," "):IF Z=0 GOTO 1150
1140 A$=LEFT$(A$,Z)+SPACE$(BL-XL)+RIGHT$(A$,LEN(A$)-Z)
1150 V=V+BL:LOCATE A,4:PRINT"↑";LEFT$(A$,35)
1160 NEXT:AC$(S,A)=A$:NEXT:LOCATE A,4:PRINT "*";LEFT$(A$,35);"  "
1170 NEXT:F$="":GOTO 300
1180 REM ROUTINE PER COPIARE UNA STROFA IN UN'ALTRA
1190 GOSUB 1530:PRINT "COPIA:":GOSUB 1360:IF M1$="F" GOTO300
1200 GOSUB 1530:PRINT"IN QUALE STROFA DEVO COPIARE QUELLA CHE VEDI ?"
1210 Q=0:INPUT "N.STROFA (RET=EXIT)";Q:IF Q=0 GOTO 300
1220 IF Q>WS+1 AND Q>MS GOTO 1200
1230 GOSUB 1500:IF A$<> "S" GOTO 300
1240 IF Q=WS+1 THEN WS=WS+1:REM CREA UNA STROFA NUOVA
1250 FOR A=1 TO MF:IF M1$="1" OR M1$="3" THENPA$(Q,A)=PA$(S,A)
1260 IF M1$="2" OR M1$="3" THEN AC$(Q,A)=AC$(S,A)
1270 NEXT:PRINT "FATTO.":GOSUB 1400:GOTO 300
1280 CLS:PRINT"PREMI TASTO 1 PER NOTAZIONE INGLESE
1290 PRINT"PREMI TASTO 2 PER NOTAZIONE ITALIANA
1300 GET A$:IF A$<>"1" AND A$<>"2" GOTO1300
1310 N=VAL(A$)-1:GOTO160
1320 REM VISUAL
1330 FOR A=1 TO MF:PRINT A;TAB(4):IF M$="N"THEN PRINT LEFT$(AC$(S,A),35)
1340 IF M$="T" THEN PRINT LEFT$(PA$(S,A),35)
1350 NEXT:RETURN
1360 Y=CSRLIN:LOCATE Y,1:PRINT"1 SOLO TESTO     2 SOLO ACCOMPAGNAMENTO
1370 LOCATE Y+1,1:PRINT "3 ENTRAMBI          F FINE
1380 GET M1$:IF M1$="F" THEN RETURN
1390 IF M1$>="1" AND M1$=<"3" THEN RETURN:ELSE GOTO 1380
1400 PRINT "PREMI UN TASTO"
1410 GET P$:IF P$="" GOTO 1410:ELSE RETURN
1420 REM MODIFICA FRASE
1430 CLS:LOCATE 5,1:PRINT "TESTO "PA$(S,F):LOCATE 5,1:LINEINPUT "TESTO ";A1$
1440 LOCATE 10,1:PRINT "ACC.  "AC$(S,F):LOCATE 10,1:LINEINPUT "ACC.  ";A2$
1450 GOSUB 1500:IFA$="N" GOTO 1430
1460 IFA$="E"THENRETURN
1470 IF A1$<>"" THENPA$(S,F)=A1$
1480 IF A2$<>"" THENAC$(S,F)=A2$
1490 RETURN
1500 PRINT"CONFERMI (S/N/EXIT) ? ";
1510 GET A$:IF A$="N" OR A$="S" OR A$="E" THEN PRINT A$:RETURN
1520 GOTO1510
1530 REM CANCELLA PARTE BASSA DELLO SCHERMO
1540 FOR A=21 TO 24:LOCATE A,1:PRINTSPACE$(40);:NEXT
1550 PRINTSPACE$(39);:LOCATE 21,1:RETURN
1560 REM SCALE MINORI
1570 DATA C-,DO-,C#-,DO#-,D-,RE-,E&-,MI&-,E-,MI-,F-,FA-,F#-,FA#-
1580 DATA G-,SOL-,G#-,SOL#-,A-,LA-,B&-,SI&-,B-,SI-
1590 REM SCALE MAGGIORI
1600 DATA C+,DO+,D&+,RE&+,D+,RE+,E&+,MI&+,E+,MI+,F+,FA+
1610 DATA G&+,SOL&+,G+,SOL+,A&+,LA&+,A+,LA+,B&+,SI&+,B+,SI+
1620 PRINT"ERRORE":PRINTEL:RESUME NEXT
1630 GOSUB 1530
1640 F$="N":PRINT "NOTAZIONE ";:IF N=0 THEN PRINT"INGLESE":ELSE PRINT"ITALIANA
1650 PRINT"VUOI CAMBIARLA ?";:GOSUB 1510:IF A$<>"S" GOTO 300
1660 IF N=1 THEN N1=1:N2=0:ELSE N1=0:N2=1
1670 GOTO 950:REM PROCEDURA QUASI COMUNE A SPOSTA TONALITA'
1680 END
```

```
NOME CANZONE 'L'ANNO CHE VERRA'' DA 1 A 3 SU 3

RE+                 SOL+
CARO AMICO TI SCRIVO
LA+7                    RE+
COSI' MI DISTRAGGO UN PO'
SI-             MI-
E SICCOME SEI MOLTO LONTANO
LA+7            RE+    LA+7
PIU' FORTE TI SCRIVERO'
RE+                  SOL+
DA QUANDO SEI PARTITO
LA+7          RE+
C'E' UNA GROSSA NOVITA'
SI-                      MI-
L'ANNO VECCHIO E' FINITO ORMAI
LA+7                    RE+
MA QUALCOSA ANCORA QUI NON VA


RE+                   SOL+
SI ESCE POCO LA SERA
LA+7                    RE+
COMPRESO QUANDO E' FESTA
SI-                          MI-
E C'E' CHI HA MESSO DEI SACCHI
       LA+7           RE+     LA+7
DI SABBIA VICINO ALLA FINESTRA
RE+                 SOL+
E SI STA SENZA PARLARE
LA+7        RE+
PER INTERE SETTIMANE
SI-                      MI-
E A QUELLI CHE HANNO NIENTE DA DIRE
LA+7        RE+
DEL TEMPO NE RIMANE


MI+           LA+
MA LA TELEVISIONE
SI+7                 MI+
HA DETTO CHE IL NUOVO ANNO
DO#-                FA#-
PORTERA' UNA TRASFORMAZIONE
SI+7                         MI+      SI+7
CHE TUTTI QUANTI STIAMO GIA' ASPETTANDO
MI+                 LA+
SARA' TRE VOLTE NATALE
SI+7           MI+
E FESTA TUTTO IL GIORNO
DO#-                  FA#-
OGNI CRISTO SCENDERA' DALLA CROCE
SI+7                  MI+
ANCHE GLI UCCELLI FARANNO RITORNO
MI+               LA+
CI SARA' DA MANGIARE
SI+7          MI+
E LUCE TUTTO L'ANNO
DO#-               FA#-
ANCHE I MUTI POTRANNO PARLARE
SI+7                MI+     SI+
MENTRE I SORDI GIA' LO FANNO
```

```
NOME CANZONE 'DUE SEMITONI IN MENO' DA 1 A 1 SU 3

DO+                 FA+
CARO AMICO TI SCRIVO
SOL+7                   DO+
COSI' MI DISTRAGGO UN PO'
LA-             RE-
E SICCOME SEI MOLTO LONTANO
SOL+7           DO+     SOL+7
PIU' FORTE TI SCRIVERO'
DO+                  FA+
DA QUANDO SEI PARTITO
SOL+7         DO+
C'E' UNA GROSSA NOVITA'
LA-                      RE-
L'ANNO VECCHIO E' FINITO ORMAI
SOL+7                   DO+
MA QUALCOSA ANCORA QUI NON VA


RE+                   SOL+
SI ESCE POCO LA SERA
LA+7                    RE+
COMPRESO QUANDO E' FESTA
SI-                          MI-
E C'E' CHI HA MESSO DEI SACCHI
       LA+7           RE+     LA+7
DI SABBIA VICINO ALLA FINESTRA
RE+                 SOL+
E SI STA SENZA PARLARE
LA+7        RE+
PER INTERE SETTIMANE
SI-                      MI-
E A QUELLI CHE HANNO NIENTE DA DIRE
LA+7        RE+
DEL TEMPO NE RIMANE


MI+           LA+
MA LA TELEVISIONE
SI+7                 MI+
HA DETTO CHE IL NUOVO ANNO
DO#-                FA#-
PORTERA' UNA TRASFORMAZIONE
SI+7                         MI+      SI+7
CHE TUTTI QUANTI STIAMO GIA' ASPETTANDO
MI+                 LA+
SARA' TRE VOLTE NATALE
SI+7           MI+
E FESTA TUTTO IL GIORNO
DO#-                  FA#-
OGNI CRISTO SCENDERA' DALLA CROCE
SI+7                  MI+
ANCHE GLI UCCELLI FARANNO RITORNO
MI+               LA+
CI SARA' DA MANGIARE
SI+7          MI+
E LUCE TUTTO L'ANNO
DO#-               FA#-
ANCHE I MUTI POTRANNO PARLARE
SI+7                MI+     SI+
MENTRE I SORDI GIA' LO FANNO
```

LETTURA DI BLOCCHI, SENZA ERRORI

Due utili routine l.m. (interamente rilocabili) per potenziare il vostro C/64

di **Gianluca Venturini**

ON ERROR GOTO

Il breve listato consente di sopperire alla mancanza, nel C/64, dell'istruzione ON ERROR GOTO presente in altri interpreti Basic.

Quando viene riscontrato un errore il computer passa ad eseguire una routine il cui indirizzo di partenza è memorizzato (nella consueta forma Lo-Hi) nelle due locazioni RAM 768 - 769 dove, di solito, sono contenuti, rispettivamente, i valori 139 e 227. Questi indicano l'inizio della routine ($E38B) che si incarica di stampare l'opportuno messaggio di errore. Giunta a questo punto, però, l'esecuzione del programma viene interrotta: per evitare che ciò accada, la nostra routine si preoccupa di modificare i vettori 768/9 in modo da farli puntare alla routine L.M. in grado di dirottare l'elaborazione, mediante un semplice GOTO, ad una linea specificata.

La sintassi della routine pubblicata è...

SYS XXXXX, Linea

...in cui "Linea" è il numero di linea da cui il programma deve continuare in caso di errore.

E' bene precisare che la routine non effettua alcun controllo sulla effettiva esistenza della linea indicata; in caso di "distrazione" il programma entra quindi in un un Loop infinito. E' peraltro possibile inserire, in uno stesso programma, più volte l'emulatore di "On Error Go To", indicando, di volta in volta, diversi numeri di linea.

Eseguite la SYS solo da programma: prima di ritornare in modo diretto rimettete a posto i vettori con:

Poke 768, 139: Poke 769, 227

Un'ultima nota di carattere tecnico: per eseguire il GOTO in L.M. è sufficiente convertire il numero di linea nei soliti valori Lo-Hi da inserire, rispettivamente, nelle locazioni 20 - 21; quindi eseguire un salto alla routine posta a partire da $A8A3.

BLOCK LOADER

Il compito di questa routine consiste nel caricare, in una pagina di memoria, il contenuto di un blocco di disco. Per far ciò la routine necessita di tre parametri...

TR, SE, SA

...in cui TR e SE indicano, rispettivamente, la traccia ed il settore del blocco di dati; SA, invece, indica la locazione di memoria a partire dalla quale vogliamo caricare il contenuto del blocco. La sintassi da seguire è...

Sys xxxx, tr, se (,sa)

...in cui il significato dei parametri è quello sopra esposto; SA è opzionale e, se omesso, assume per default il valore 1024 (il contenuto del blocco viene visualizzato sullo schermo a partire dalla prima cella video).

Il disassemblato è sufficientemente chiaro per capire come caricare un blocco in L.M; le routines usate sono quelle del KERNAL. Qui di seguito è riportata la spiegazione del funzionamento di alcune di esse:

LISTEN
Predispone una periferica a ricevere dati dal bus seriale.

Chiamata: FFB1
Comunicazione: reg. A

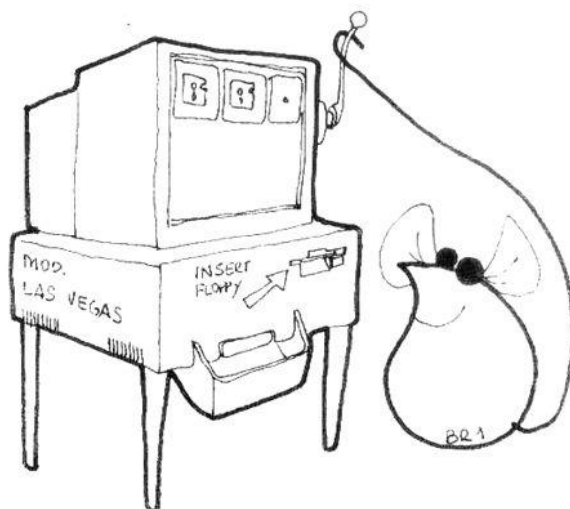
Prima della chiamata si deve porre in A un numero compreso tra 4 e 30; esso viene decodificato per ottenere l'indirizzo della periferica; questo le viene inviato e la pone in stato di attesa.

SECOND
Invia l'indirizzo secondario (Secondary Address).

Chiamata: FF93
Comunicazione: reg. A
Preparazione: Listen

Deve essere usata dopo LISTEN se per il dispositivo è necessario un indirizzo secondario. Se il dispositivo è sul bus seriale, l'indirizzo secondario deve essere modificato con OR 96 ($60) prima di usarlo; deve essere posto in A.

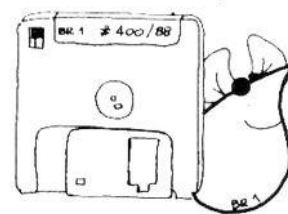
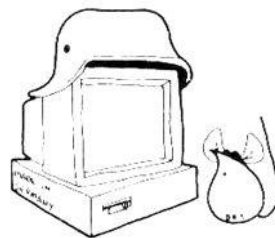
CIOUT
Invia un byte sul bus seriale.

Chiamata:FFA8

Comunicazione: reg. A
Preparazione: Listen, Second

Il dispositivo deve essere in stato LISTEN, altrimenti si ha errore di timeout. Se risulta necessaria l'indicazione dell'indirizzo secondario, deve essere dapprima attivata la routine SECOND.

Il carattere da inviare deve essere posto in A. La routine mantiene un carattere nel buffer. Quando viene chiamata UNLSN viene inviato il carattere presente nel buffer insieme a EOI e il dispositivo esce dallo stato di attesa.

Il breve programma "Basic block loader" altri non è che il corrispondente BASIC della routine L.M.

```
100 REM CARICATORE ROUTINE L.M. PER C/64
110 :
120 X=49151:Y=0
130 READ A:IF A<0 THEN READ B:GOTO 150
140 POKE X+1,A:Y=Y+A:X=X+1:GOTO130
150 IF B<> Y THEN PRINT"ERRORE NEI DATA":END
160 GOSUB 1000:END
170 :
1000 PRINT CHR$(147);"ON ERROR GOTO":PRINT
1010 PRINT"USO: SYS XXXX,NUMERO LINEA":PRINT
1020 PRINT"XXXX=INDIRIZZO DI PARTENZA QUALSIASI":PRINT
1030 PRINT"NUMERO LINEA= - DA CUI DEVE CONTINUARE                 ";
1040 PRINT"IL PROGRAMMA IN CASO                  DI ERRORE":PRINT
1050 PRINT" RICORDA: POKE 768,139:POKE 769,227  PRIMA DI USCIRE DAL";
1060 PRINT" PROGRAMMA":PRINT
1070 RETURN
1100 DATA 165,020,133,254,165,021,133,255,032,253
1110 DATA 174,032,138,173,032,247,183,160,048,165
1120 DATA 020,145,254,160,052,165,021,145,254,024
1130 DATA 169,047,101,254,141,000,003,144,002,230
1140 DATA 255,165,255,141,001,003,096,169,000,133
1150 DATA 020,169,000,133,021,076,163,168,-1,7082
1160 END
```

UN'ENCICLOPEDIA PER IL TUO COMMODORE?

Straordinario!

Commodore Computer Club ti offre un'eccezionale combinazione, valida fino ad esaurimento delle scorte.

Ai nuovi lettori, infatti, offre gli 11 fascicoli dell'intera annata 1988, dal n.49 (gennaio) al n.59 (dicembre) inclusi, al favoloso prezzo di L. 49000, spese di spedizione comprese.

Per usufruire della fantastica offerta, invia subito la cifra richiesta mediante assegno bancario (non trasferibile), intestato a:

Systems Editoriale
Servizio arretrati
Viale Famagosta, 75
20142 MILANO

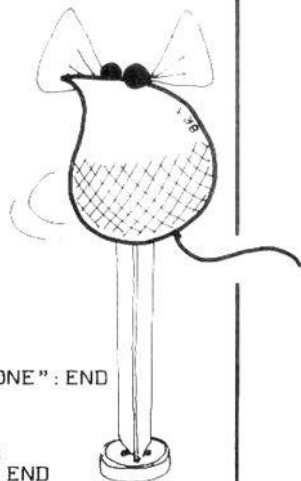
A causa dell'eccezionalità dell'offerta, gli abbonati non possono usufruire del consueto sconto a loro riservato.

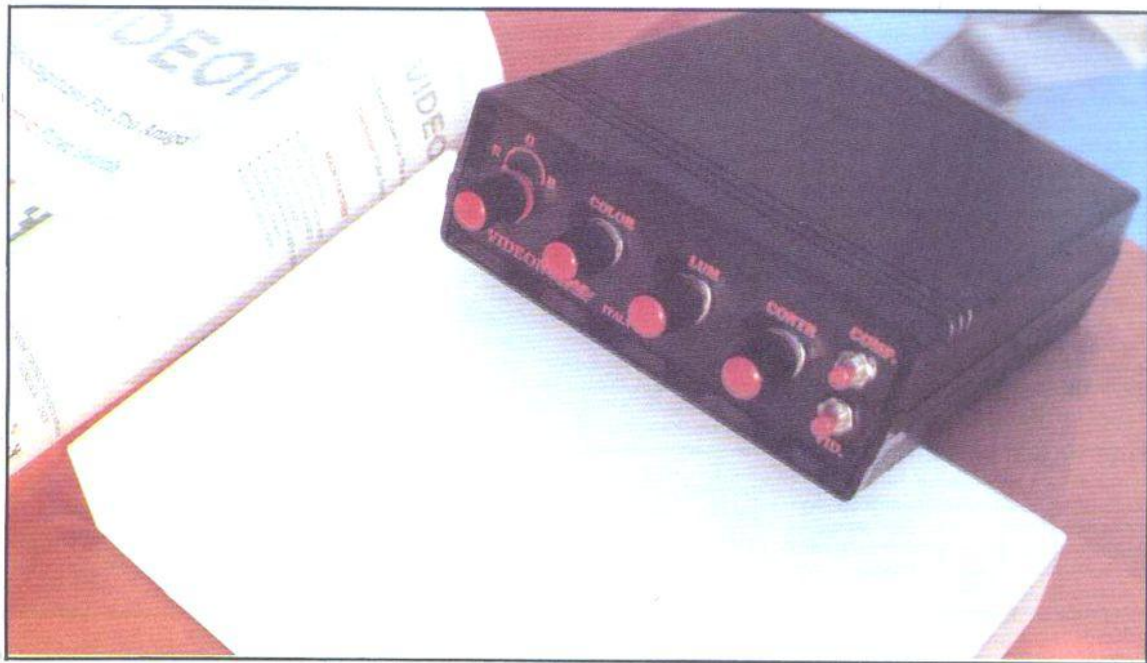
Ricorda che i numeri arretrati di una pubblicazione tecnica, come la nostra, sono spesso più utili di una "vera" enciclopedia: articoli di informazione generale, risposte ai lettori, listati, tecniche di programmazione, didattica, utility, giochi... sono soltanto alcuni degli argomenti costantemente affrontati sulle pagine di "Commodore Computer Club" e risultano perfettamente comprensibili ai principianti ed utilissimi agli utenti evoluti.

N.B.: dal momento che i numeri arretrati sono in fase di rapido esaurimento, ricorda di indicare, nella lettera di accompagnamento, i numeri dei fascicoli alternativi che desideri, nel caso non sia possibile inviarti l'intera annata.

```
10 REM * DEMO PER ON ERROR GOTO
20 REM ** BY GIANLUCA VENTURINI
30 :
60 X=49152:REM START ADDRESS !!!
70 IFX=0 THEN PRINT"LINEA 60!":END
80 SYS X,110
90 REM GENERA UN ERRORE
100 FOR I=10 TO 0 STEP-1
105 PRINT"1/"I"="1/I:NEXTI
110 PRINT"ERRORI IN 130, 160"
120 SYS X,140
130 GOTO1000 :REM LINEA INESISTENTE
140 PRINT"PRIMA INTERCETTAZIONE"
150 SYS X,170
160 A$=1 :REM TYPE MISMATCH
170 PRINT"SECONDA INTERCETTAZIONE"
180 POKE768,139:POKE769,227
190 PRINT"NOTA BENE:": LIST 180: END
```

```
10 REM CORRISPONDENTE  BASIC DELLA
15 REM ROUTINE 'BLOCK LOADER' L.M.
20 REM ***************************
25 REM *****GIANLUCA VENTURINI****
30 REM ***************************
35 INPUT"TRACCIA";TR:IFTR<1ORTR>35THEN35
40 INPUT"SETTORE";SE:IFSE<0ORSE>20THEN40
50 SA=1024:INPUT"START ADDRESS";SA
60 OPEN3,8,3,"#"
70 OPEN15,8,15:PRINT#15,"U1:";3;0;TR;SE
80 PRINT#15,"B-P 3 0"
90 FORI=0TO255:GET#3,A$:IFA$=""THENA$=CHR$(0)
100 IFSTTHEN120
110 POKESA+I,ASC(A$):NEXT
120 CLOSE3:PRINT#15,"I0":CLOSE15
```

```
100 REM BLOCK READER. C/64 + DISK DRIVE
110 :
180 Y=-1:GOSUB1000:PRINT:INPUT"DA QUALE LOCAZIONE";X
190 READW:Y=Y+1:IFW<0THEN210
200 GOTO190
210 PRINT"PRIMA LOCAZIONE ="X
220 PRINT"ULTIMA LOCAZIONE ="X+Y-1:PRINT
230 PRINT"CONFERMI? (S/N)"
240 GETA$:IFA$=""THEN240
250 IFA$="S"THENRESTORE:T=X:GOTO280
260 RUN
270 :
280 GOSUB1000:PRINT:PRINT"ATTENDERE...":PRINT:W=0
290 READB:IFB>=0THENPOKET,B:T=T+1:W=W+B:GOTO290
300 READB:IFB<>WTHENPRINT:PRINTCHR$(18)"ERRORE DI TRASCRIZIONE":END
310 PRINT"ROUTINE ALLOCATA DA"X"A"T-1"COMPR.":PRINT
320 PRINT"ATTIVARE CON SYS"X:PRINT
330 X1=INT(X/256):X2=X-X1*256:PRINT"POKE 44,"X1":POKE43,"X2;
340 Y1=INT(T/256):Y2=T-Y1*256:PRINT":POKE46,"Y1":POKE45,"Y2:END
1000 PRINT CHR$(147);"BLOCK LOADER":PRINT
1010 PRINT"USO: SYS XXXX,TR,SE [,SA]":PRINT
1020 PRINT"XXXX=INDIRIZZO DI PARTENZA QUALSIASI":PRINT
1030 PRINT"TR=NUMERO TRACCIA (1..35)":PRINT
1040 PRINT"SE=NUMERO SETTORE (0..20)":PRINT
1050 PRINT"SA=START ADDRESS(OPZIONALE):DEFAULT=1024":PRINT
1060 RETURN
1100 DATA 056,176,015,035,085,049,058,051,032,048
1110 DATA 032,066,045,080,032,051,032,048,165,020
1120 DATA 133,142,165,021,133,143,032,241,183,224
1130 DATA 000,208,003,076,072,178,224,036,016,249
1140 DATA 134,254,032,241,183,224,021,016,240,134
1150 DATA 255,169,000,133,251,166,251,181,254,170
1160 DATA 169,048,133,252,169,047,133,253,232,230
1170 DATA 253,165,253,201,058,208,006,230,252,169
1180 DATA 048,133,253,202,208,239,165,251,208,008
1190 DATA 165,252,133,140,165,253,133,141,230,251
1200 DATA 165,251,201,002,208,205,169,000,133,003
1210 DATA 169,004,133,004,160,000,177,122,201,044
1220 DATA 208,017,032,253,174,032,138,173,032,247
1230 DATA 183,165,020,133,003,165,021,133,004,169
1240 DATA 003,162,008,160,003,032,186,255,169,001
1250 DATA 166,142,164,143,232,208,001,200,232,208
1260 DATA 001,200,232,208,001,200,032,189,255,032
1270 DATA 192,255,169,008,032,177,255,169,111,032
1280 DATA 147,255,160,004,177,142,032,168,255,200
1290 DATA 192,011,208,246,165,140,032,168,255,165
1300 DATA 141,032,168,255,169,032,032,168,255,165
1310 DATA 252,032,168,255,165,253,032,168,255,032
1320 DATA 174,255,169,008,032,177,255,169,111,032
1330 DATA 147,255,177,142,032,168,255,200,192,018
1340 DATA 208,246,169,000,032,168,255,032,174,255
1350 DATA 162,003,032,198,255,160,000,032,207,255
1360 DATA 145,003,165,144,208,003,200,208,244,032
1370 DATA 204,255,169,008,032,177,255,169,111,032
1380 DATA 147,255,169,073,032,168,255,169,048,032
1390 DATA 168,255,032,174,255,032,231,255,096,-1,41653
1400 END
```

```
DISASSEMBLATO ROUTINE:
1
2 * ON ERROR GOTO : SYS XXXX, LINEA
3
4            ORG   49152
5  INIZIO    LDA   $14       ;#<INIZIO (VEDI CCC)
6            STA   $FE       ;IN 254 ; (N.46      )
7            LDA   $15       ;#>INIZIO (PAG. 30 )
8            STA   $FF       ;IN 255.
9            JSR   $AEFD     ;PRENDE UNA VIRGOLA
10           JSR   $ADBA     ;E IL NUMERO DI LINEA
11           JSR   $B7F7     ;E LO MEMOR. IN 20-21
12           LDY   #$30      ;METTE IL
13           LDA   $14       ;NUMERO DI LINEA (LO)
14           STA   ($FE),Y   ;IN START+1
15           LDY   #$34      ;METTE IL
16           LDA   $15       ;NUMERO DI LINEA (HI)
17           STA   ($FE),Y   ;IN START+5
18           CLC             ;MODIFICA IL VETTORE IN
19           LDA   #$2F      ;MODO DA FARLO PUNTARE
20           ADC   $FE       ;ALLA ROUTINE 'START'
21           STA   $0300
22           BCC   SALTO
23           INC   $FF
24 SALTO     LDA   $FF
25           STA   $0301
26           RTS
27 START     LDA   #$00      ;(00=VALORE FITTIZIO )
28           STA   $14       ;CARICA LE LOCAZIONI
29           LDA   #$00      ;20 - 21 CON IL NUMERO
30           STA   $15       ;DI LINEA ED INFINE
31           JMP   $A8A3     ;ESEGUE IL GOTO
```

```
DISASSEMBLATO COMMENTATO BLOCK LOADER

1  SETLFS    =     $FFBA
2  SETNAM    =     $FFBD
3  LISTEN    =     $FFB1
4  SECOND    =     $FF93
5  CIOUT     =     $FFA8
6  UNLSN     =     $FFAE
7  CHKIN     =     $FFC6
8  CHRIN     =     $FFCF
9  CLRCHN    =     $FFCC
10 CLALL     =     $FFE7
11 *******************************
12 * BLOCK LOADER                *
13 * SYS XXXX, TR, SE (,SA)      *
14 *******************************
15 * BY  GIANLUCA VENTURINI      *
16 *                             *
17 *******************************
18           ORG   49152
19 START     SEC
20           BCS   INIZIO
21 TEXT      TXT   '#U1 3 Ø B-P 3 Ø'
22 INIZIO    LDA   $14       ;#<START
23           STA   $8E       ;in 142
24           LDA   $15       ;#>START
25           STA   $8F       ;in 143
26           JSR   $B7F1     ;prende un parametro a 8 bit in X
27           CPX   #$00      ;e lo confronta:
28           BNE   SALTO1    ;se <=0
29 ERR       JMP   $B248     ;stampa 'ILLEGAL QUANTITY'
30 SALTO1    CPX   #$24      ;se >35
31           BPL   ERR       ;errore come sopra
32           STX   $FE       ;pone TR in 254
33           JSR   $B7F1     ;prende un parametro a 8 bit in X
34           CPX   #$15      ;se >=21
35           BPL   ERR       ;errore come prima
36           STX   $FF       ;pone SE in 255
37           LDA   #$00      ;questa parte si occupa
38           STA   $FB       ;di convertire i valori
39 SALTO2    LDX   $FB       ;TR e SE in coppie di
40           LDA   $FE,X     ;valori che rappresentano
41           TAX             ;le cifre ASCII dei due
42           LDA   #$30      ;parametri per poterli
43           STA   $FC       ;inviare sul bus seriale.
44           LDA   #$2F      ;esempio : 10 =49 (1) - 48 (0)
45           STA   $FD
46           INX
47 SALTO3    INC   $FD
48           LDA   $FD
49           CMP   #$3A
50           BNE   SALTO4
51           INC   $FC
52           LDA   #$30
53           STA   $FD
54 SALTO4    DEX
55           BNE   SALTO3
56           LDA   $FB
57           BNE   SALTO5
58           LDA   $FC
59           STA   $8C
60           LDA   $FD
61           STA   $8D
62 SALTO5    INC   $FB
63           LDA   $FB
64           CMP   #$02
65           BNE   SALTO2
66           LDA   #$00      ;default SA=1024
67           STA   $03
68           LDA   #$04
69           STA   $04
70           LDY   #$00
71           LDA   ($7A),Y   ;il prossimo carattere
72           CMP   #$2C      ;e' una virgola ?
73           BNE   SALTO6    ;NO  SA= default
74           JSR   $AEFD     ;SI  prende virgola,
75           JSR   $ADBA     ;prende SA
76           JSR   $B7F7     ;e lo pone in $14-15
77           LDA   $14       ;copia $14-15
78           STA   $03       ;in $03-04
79           LDA   $15
80           STA   $04
81 SALTO6    LDA   #$03      ;carica parametri
82           LDX   #$08      ;per l'apertura
83           LDY   #$03      ;del file
84           JSR   SETLFS    ;e li setta
85           LDA   #$01      ;Filename length=1
86           LDX   $8E       ;alla fine ho:
87           LDY   $8F       ;X= #<TEXT
88           INX             ;Y= #>TEXT
89           BNE   SALTO7
90           INY
91 SALTO7    INX
92           BNE   SALTO8
93           INY
94 SALTO8    INX
95           BNE   SALTO9
96           INY
97 SALTO9    JSR   SETNAM    ;file con nome '#'
98           JSR   OPEN      ;open 3,8,3,"#"
99           LDA   #$08      ;prepara la
100          JSR   LISTEN    ;periferica 8
101          LDA   #$6F      ;indirizzo
102          JSR   SECOND    ;secondario 15
103          LDY   #$04
104 SALTO10  LDA   ($8E),Y   ;e invia i caratteri
105          JSR   CIOUT     ;'U1 3 Ø'
106          INY
107          CPY   #$08
108          BNE   SALTO10
109          LDA   $8C       ;invia le due cifre
110          JSR   CIOUT     ;di TR
111          LDA   $8D
112          JSR   CIOUT
113          LDA   #$20      ;uno spazio
114          JSR   CIOUT
115          LDA   $FC       ;quindi le due
116          JSR   CIOUT     ;cifre di SE
117          LDA   $FD
118          JSR   CIOUT
119          JSR   UNLSN     ;chiude
120          LDA   #$08
121          JSR   LISTEN    ;e riapre
122          LDA   #$6F
123          JSR   SECOND    ;il canale
124 SALTO11  LDA   ($8E),Y   ;per inviare
125          JSR   $FFA8     ;'B-P 3 Ø'
126          INY
127          CPY   #$12
128          BNE   SALTO11
129          LDA   #$00
130          JSR   CIOUT
131          JSR   UNLSN     ;unlisten
132          LDX   #$03
133          JSR   CHKIN     ;canale 3 di lettura
134          LDY   #$00
135 SALTO12  JSR   CHRIN     ;prende uno ad uno i bytes
136          STA   ($03),Y   ;li scrive a partire da SA
137          LDA   $90       ;legge status byte
138          BNE   SALTO13   ;se diverso da zero esce
139          INY
140          BNE   SALTO12   ;ripete 256 volte
141 SALTO13  JSR   CLRCHN    ;chiude tutti i canali
142          LDA   #$08
143          JSR   LISTEN
144          LDA   #$6F
145          JSR   SECOND
146          LDA   #$49      ;'I'
147          JSR   CIOUT
148          LDA   #$30      ;'Ø'
149          JSR   CIOUT
150          JSR   UNLSN     ;invia 'IØ' al drive
151          JSR   CLALL     ;chiude tutti i file
152          RTS             ;e torna al basic.
```

FOTOGRAFARE CON AMIGA

Un digitalizzatore video è proprio quello che ci vuole per elaborare le immagini riprese con telecamera o videoregistratore

di **Stefano Simonelli**

Quasi tutti, ormai, possiedono un home computer corredato da un monitor dedicato alla visualizzazione delle immagini.

Non molto tempo fa, quando ancora non esisteva la potente periferica che oggi è diffusissima, si utilizzavano terminali telescriventi privi di monitor, i quali potevano comunicare, al codiddetto ambiente "esterno", il risultato delle elaborazioni svolte solo attraverso una lunga (e lenta) stampa su carta.

E' facile intuire che le operazioni da svolgere, connesse agli sprechi di tempo che si venivano ad accumulare, rendevano proibitivo l'uso del calcolatore per applicazioni particolari.

Oggi, per fortuna, i tempi sono cambiati. Il monitor è un apparecchio tanto comune da venir considerato parte integrante dello stesso computer. Esiste, tuttavia, una notevole differenza tra i segnali analogici del video (le cui grandezze variano un po' per volta, linearmente) ed il sistema "tutto o niente" del mondo digitale (che assume due soli valori, zero oppure uno, "On" oppure "Off" e così via) cui appartiene il computer; pochi si soffermano, però, sulle differenze che intercorrono tra i due sistemi di visualizzazione.

IL VIDEO, COME FUNZIONA

La televisione ebbe un grandissimo successo quando fu presentata sul mercato e continue ricerche vennero intraprese per ridurre i costi di produzione in serie dei tubi catodici. E' solo negli ultimi anni, però, che le ricerche hanno avuto, come obiettivo, lo studio di apparecchi video dedicati esclusivamente all'utilizzo nel campo dell'informatica.

Prima di procedere non è superflua una breve introduzione sul funzionamento dell'immagine video in generale.

La maggior parte degli schermi attualmente prodotti utilizza un tubo a raggi catodici ed un sistema detto "raster scan". La superficie anteriore interna di un televisore (o monitor) è ricoperta di fosforo, mentre, in posizione arretrata, è presente un generatore di elettroni che li "spara" (è proprio questo il termine usato nei manuali tecnici) contro lo schermo frontale. Al momento dell'impatto tra gli elettroni ed il fosforo, quest'ultimo si illumina per una frazione di secondo ed il continuo movimento del fascio, sul video, genera un'immagine; la sequenza di immagini singole, ma visualizzate ad alta velocità, riproduce l'effetto dell'animazione, allo stesso modo dei fotogrammi di un qualsiasi film. Il "raster scan" è il sistema che controlla e indirizza l'emissione degli elettroni idonei a generare l'immagine che si desidera.

Il segnale che comunica al pennello (così viene spesso chiamato il fascio di elettro-

ni) il movimento da effettuare è un'unica onda sinusoidale, privo di interruzioni. Il computer, al contrario (come pure qualsiasi componente ad esso collegato, tra cui il circuito video) viene gestito da una sequenza di valori "1" e "0", caratterizzata da pause.

L'ostacolo maggiore è quindi costituito dalla necessitàdi trasformare i dati presenti nel computer in segnali video, idonei ad esser "trattati" mediante tecniche ben note. La soluzione al problema fu rappresentata dalla creazione dei convertitori digitali/analogici (D/A) grazie ai quali si eliminano gli intervalli tipici del mondo digitale, "riempendo" gli spazi di tempo inutilizzati in modo da originare un segnale continuo che il monitor è in grado di visualizzare.

DAL VIDEO AL COMPUTER

In questi ultimi anni l'evoluzione tecnologica ha reso possibile il processo inverso, vale a dire che è possibile immettere nella memoria del computer immagini provenienti da un videoregistratore, una telecamera o da un altro computer.

Il processo è conosciuto con il nome di "video - digitalizzazione", che si realizza utilizzando un apparecchio chiamato (guarda caso) digitalizzatore. La "chiave" del meccanismo è costituita da un convertitore analogico-digitale (A/D) che, come è facile immaginare, svolge il compito di trasformare in una sequenza di bit il segnale video in relazione ad altri due parametri molto importanti che il circuito A/D deve rispettare: la posizione ed il colore di ciascun pixel dello schermo.

Il processo per determinare il colore di un singolo pixel risulta piuttosto complicato, ma cercheremo di esporlo nel modo più semplice possibile.

DAL COLORE AL BIT

E' noto che un qualsiasi colore può esser considerato come una particolare combinazione dei tre colori fondamentali: il ROSSO, il VERDE e il BLU. Di conseguenza si può comporre qualsiasi colore (comprese varie caratteristiche, come luminosità, sfumature e così via) conoscendo le percentuali dei tre colori; ed è proprio questa la funzione che un digitalizzatore svolge, grazie, soprattutto, all'azione di alcuni filtri.

Questi permettono di ottenere, elaborando un'immagne colorata, tre immagini contenenti, rispettivamente, solo la sua componente rossa, quella verde e quella blu. Con il primo filtro, in pratica, viene privilegiata la "trasmissione" del colore rosso (e delle sue "componenti") mentre rimangono "bloccate" tutte le altre.

La tecnica appena descritta viene spesso usata dai fotografi che usano pellicole in bianco e nero: ponendo sull'obiettivo fotografico un filtro rosso, un oggetto colorato in rosso si "schiarisce" mentre oggetti blu tendono a scurirsi. Il contrario accade ponendo un filtro blu. E' ovvio che fotografando uno stesso oggetto tre volte (cambiando filtro di volta in volta) ed elaborando opportunamente i tre fotogrammi ottenuti, sarà possibile stabilire, per ciascun oggetto ripreso, le quantità di colore rosso, blu e verde possedute.

Sincronizzando il computer con l'emissione del segnale video, viene comparata la percentuale di rosso, di ciascun pixel, con alcuni dati di riferimento allo scopo di determinarne il valore da memorizzare. L'operazione, in pratica, si effettua calcolando la tensione del segnale nel momento in cui il pixel stesso viene inviato. Terminata l'operazione di filtraggio del rosso (setacciando ciascun pixel dell'intera immagine video) si passa ad eseguirla anche per le componenti verde e blu. Alla fine, in memoria, saranno presenti tre mappe, una per il rosso una per il verde e l'ultima per il blu.

Solo a questo punto, e rispettando le caratteristiche hardware proprie del computer, sarà possibile miscelare opportunamente le tre percentuali relative ad uno stesso pixel in modo da ottenere il colore definitivo.

LA CATTURA

L'immagine proveniente dall'apparecchiatura esterna, per esser "catturata", deve risultare perfettamente immobile, almeno durante l'arco di tempo occorrente per scandire la videata tre volte, una per ciascun filtraggio. La procedura trova pertanto i suoi limiti nella ripresa di immagini che si muovono con grande rapidità.

Determinare (in tempo reale) il colore e la posizione di ogni pixel è un'operazione piuttosto lunga. Per digitalizzare un'immagine sono infatti necessari alcune decine di secondi, indispensabili per svolgere la miriade di calcoli a notevole velocità.

LE IMMAGINI DI AMIGA

Abbiamo avuto modo di provare un video-digitalizzatore per Amiga messo gentilmente a nostra disposizione dalla Ditta "Flopperia" di Milano (Via Monte Nero, 31 tel: 02/55.18.04.84); di seguito riportiamo le caratteristiche salienti emerse dalla prova.

"Videon" è un digitalizzatore a colori che permette di digitalizzare l'immagine proveniente da una sorgente PAL videocomposita (cioè lo standard Europeo) come una telecamera, un videoregistratore (provvisto di fermo immagine di precisione) o un computer.

Il software allegato alla confezione è in grado di acquisire immagini nelle varie risoluzioni proprie di Amiga: 640 X 256, 320 X 256. Con il modo interlace la risoluzione grafica passa a 640 X 512 pixel oppure a 320 X 512; è presente anche la modalità "overscan" che ha il pregio di eliminare i bordi (molto utile per usi professionali) e di manipolare le immagini con tecniche di SURFACE MAPPING per salvarle su disco in modo da rielaborarle con programmi di grafica. Una volta catturata l'immagine è infatti possibile "ritoccarla" con programmi specifici, come Photon Paint o Delux Paint. Ci si può quindi divertire a modificare forme e colori o a riprodurre più particolari di una stessa immagine.

Nella confezione sono presenti diversi accessori: un alimentatore da 15 Volt, l'indispensabile cavo bypass per il monitor, un cavo di collegamento Videon-computer, il dischetto contenente il software dedicato, la garanzia ed il manuale, che spiega in modo dettagliato sia l'installazione sia le modalità d'uso del Videon.

Videon è un parallelepipedo di metallo dotato (sul retro) degli opportuni connettori per i vari collegamenti. Sul frontalino, invece, è presente una manopola a tre posizioni contrassegnate dalle indicazioni Rosso, Verde, Blu (per filtrare il segnale). Due pulsanti (necessari per bypassare il segnale da digitalizzare senza bisogno di un secondo monitor) consentono la selezione dell'immagine da visualizzare (quella "sorgente" o il video di Amiga). Tre manopole, (COLOR, LUM e CONTR) consentono infine di regolare, rispettivamente, la saturazione del colore, la luminosità ed il contrasto dell'immagine sorgente.

IL SOFTWARE

Collegati i cavi (a computer SPENTO) e riacceso Amiga, apparirà la schermata principale grazie alla quale potrete, a scelta, iniziare la digitalizzazione, selezionare il tipo di risoluzione, salvare su disco l'immagine "catturata" o manipolarla con la tecnica di SURFACE MAPPING. Grazie a quest'ultima sarete in grado, in modo del tutto automatico, di "arrotolare" l'immagine sulla superficie di un solido tridimensionale, le cui forme (previste dal programma) sono davvero numerose; si possono perfino alterare le dimensioni e la prospettiva! I risultati, statene certi, non mancheranno di stupirvi.

Una volta scelta la risoluzione desiderata non resterà che passare alla digitalizzazione vera e propria, selezionando (con la manopola R, G, B) la lettera R e clickando con il mouse sul riquadro Red (rosso).

Dopo una pausa di qualche secondo, comparirà un'immagine in bianco e nero, corrispondente alla componente rossa dell'immagine finale. Commutando poi su "G" e, successivamente, su "B", si ripete l'operazione al termine della quale (selezionando Display) viene visualizzata l'immagine definitiva.

CONCLUSIONI

Digitalizzare non è un processo da affidare solo a pochi eletti; non solo, infatti, è facile, ma i risultati sono stupefacenti.

Una volta elaborata l'immagine, ed eventualmente perfezionata, sarà possibile aggiungere il disegno a vostri programmi, così da conferire il tocco finale al lavoro.

Il giudizio relativo alla periferica non può essere che positivo, essendo questa "completa" ed utilizzabile anche per scopi semiprofessionali. L'installazione, inoltre, è semplice e banale; l'unico neo è forse rappresentato dal fatto che, per caricare il software, è necessaria l'espansione da almeno 512 KRam.

E' pur vero, tuttavia, che quasi tutti i possessori di questo straordinario calcolatore hanno già provveduto, da tempo, a procurarsi sia il drive aggiuntivo sia l'espansione da (almeno) 500 K.

DIFFERENZE

Le schede realizzate in precedenza per Amiga (e per altri computer) utilizzavano una telecamera in bianco e nero che, opportunamente collegata al calcolatore, svolgeva il compito di riprendere le immagini da digitalizzare.

Il filtraggio dei tre colori veniva realizzato in modo piuttosto rudimentale dal momento che, tenendo il soggetto perfettamente immobile, era necessario effettuare tre riprese distinte durante le quali sovrapporre, sull'obiettivo della telecamera stessa, altrettanti filtri fotografici (verde, blu e rosso), indispensabili per la necessaria selezione dei colori.

Il metodo, inutile dirlo, risultava piuttosto scomodo anche perchè è sufficiente un minimo movimento inopportuno, soprattutto durante la fase del cambio del filtro, per "disassare" le tre immagini, con conseguente perdita di nitidezza.

"Videon", al contrario, realizza il fisltraggio in modo elettronico, elaborando direttamente il segnale video che, quindi, può provenire da una telecamera a colori, da un videoregistratore o da un TV color opportunamente dotato di connettore.

DUE PROGRAMMI DI UN CERTO CARATTERE

Pubblichiamo volentieri alcuni listati inviati da un nostro lettore 128-ista

di **Alessio Palma**

Il programma "Megaprint", scritto completamente in linguaggio macchina, permette di stampare maxi caratteri e funziona sul C/128 in modalità 40 oppure 80 colonne. Nel caso venga usato quest'ultimo è consigliabile attivare il modo Fast perchè, in caso contrario (sebbene la routine si in l.m.) la visualizzazione risulterebbe piuttosto lenta.

Sono state usate le routine del S.O. che, purtroppo, hanno mostrato i loro limiti nello scrivere sullo schermo ad 80 colonne; ciò consente, tuttavia, di utilizzare il programma sia in modalità 80 che 40 colonne, senza apportare modifiche di sorta.

FONTASTIC

Capita molto spesso di ridefinire un set di caratteri e di volerlo usare, in seguito, anche in ambiente Basic. Purtroppo caricare un set di caratteri, ed abilitarlo, si rivela un'operazione piuttosto atroce, specie se lo si vuole abilitare sullo schermo ad 80 colonne.

Il programma "Fontastic" carica il set, da disco, e lo pone nella pagina grafica; in base all'individuazione (automatica) del modo video (80 oppure 40 colonne) agisce di conseguenza.

Nel caso delle 80 colonne il set viene scritto nella Dram; nel caso delle 40 colonne, invece, il set di caratteri occuperà la zo-

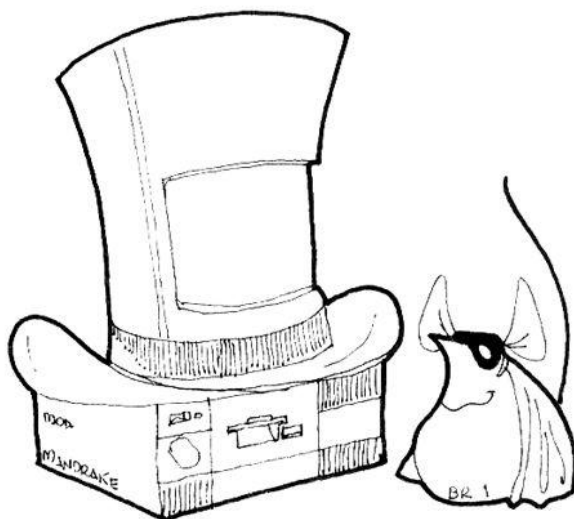
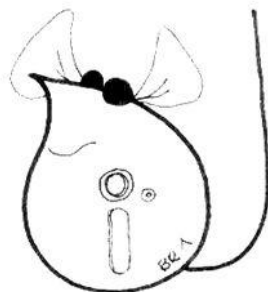
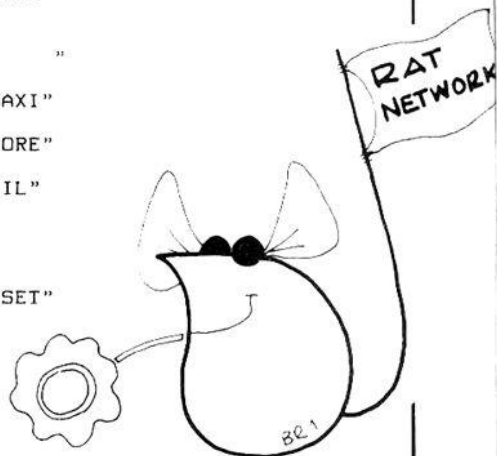
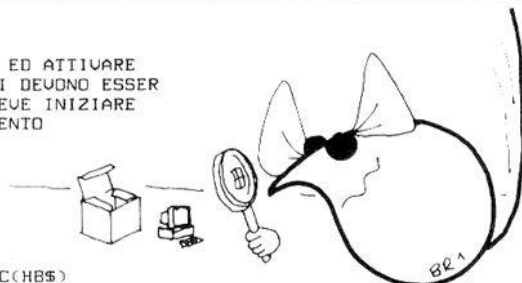
na dedicata alla pagina grafica. Non sarà quindi possibile, in quest'ultimo caso, usare la pagina grafica. Per evitare inconvenienti, il tasto F1, che normalmente attiva il comando "GRAPHIC" verrà rimpiazzato da un nuovo comando (POKE 2604, 24) idoneo a rendere nuovamente operativo il set di caratteri nel caso vengano premuti i tasti Run/Stop + Restore.

COME FUNZIONA

E' inutile ricordare che il programma può funzionare solo se l'utente dispone già di altri set di caratteri alternativi. Sono molti, del resto, i set di pubblico dominio attualmente in circolazione: alcuni vengono periodicamente diffusi sul nostro dischetto "Directory", in vendita per corrispondenza.

"Fontastic" scriverà a seconda del set selezionato: Doppio o Singolo.

Ciò significa che il set ridefinisce tutti i caratteri, anche quelli che si ottengono premendo Shift ed il tasto Commodore.

```
1 REM PROGRAMMA PER GESTIRE MESSAGGI GIGANTI
2 REM SULLO SCHERMO DEL C/128 (40 OPPURE 80 COLONNE)
3 REM INDISPENSABILE CARICARE PRIMA IL PROGRAMMA LM
4 GRAPHIC 5 :REM SCHERMO A 80 COLONNE
5 Z=-4:SCNCLR:C=64
10 READA$,S,F:IFA$="*" THEN END
30 Z-Z+4:IF Z>18 OR F-1 THENGETKEYK$:PRINT"□":Z=0
40 FOR J-1 TO LEN(A$)
50 P$-MID$(A$,J,1)
60 IF ASC(P$)<64 THEN P$-CHR$(ASC(P$)+C)
70 CHAR1,0,Z,"":SYS3028,ASC(P$)-C,(J-1)*4,S
80 NEXT
90 GOTO 10
95 DATA" ",0,0
96 DATA" ",0,0
100 DATA"QUESTO E' MEGAPRINT",0,0
110 DATA"IDEATO DA P.ALESSIO",0,0
120 DATA"CON QUESTA ROUTINE",0,1
130 DATA"SI PUO' FACILMENTE",0,0
140 DATA"REALIZZARE UNA  ",0,0
150 DATA"PRESENTAZIONE",0,0
160 DATA"OPPURE MOSTRARE ",0,1
170 DATA"UN MESSAGGIO CHE",0,0
180 DATA"DEVE ESSERE LETTO",0,0
190 DATA"AD OGNI COSTO...",0,0
200 DATA" ",0,1
210 DATA" ",0,0
220 DATA"     LEGGIMI !!",0,0
230 DATA" ",0,0
240 DATA" ",0,0
250 DATA"NATURALMENTE PUOI",0,1
260 DATA"UTILIZZARE ANCHE",0,0
270 DATA"CARATTERI MINUSCOLI",1,0
280 DATA"E NUMERI 12938",0,0
290 DATA" ",0,1
300 DATA"IN 80 COLONNE SI",0,0
310 DATA"HANNO 20 X 10 -",0,0
320 DATA"200 CARATTERI..",0,0
330 DATA"SE POSSEDEVI UN",0,0
340 DATA"VIC-20 NON TI ",0,0
350 DATA"SEMBRERANNO POCHI",0,0
360 DATA" ",0,1
370 DATA"E' TUTTO .......",0,0
380 DATA"..ARRIVEDERCI",0,0,*,0,0
390 END
```

```
100 REM +----------------------------------+
110 REM !                                  !
120 REM ! M E G A     P R I N T    C/128   !
130 REM ! ------------------------------   !
140 REM !  (C)   BY       ALESSIO PALMA    !
150 REM !                                  !
160 REM +----------------------------------+
170 :
180 FOR I=2816 TO 3073:READ II:POKEI,II:CK=CK+II:NEXT
190 IF CK<>30024 THEN PRINT"DATI ERRATI !!!":END
200 PRINT""
210 PRINT" MEGA PRINT DI P.ALESSIO.
220 PRINT
230 PRINT"    QUESTA ROUTINE PERMETTE DI STAMPARE MAXI"
240 PRINT"    CARATTERI SULLO SCHERMO."
250 PRINT"    ESSA RISIEDE NEL BUFFER DEL REGISTRATORE"
260 PRINT"    SI CANCELLA OGNI VOLTA SI CARICA UN"
270 PRINT"    PROGRAMMA DA NASTRO,OPPURE SI ESEGUE IL"
280 PRINT"    BOOT DA DISCO."
290 PRINT
300 PRINT"-SINTASSI:"
310 PRINT
320 PRINT"    SYS 3028,CODICE_CARATTER,POSIZIONE_X,SET"
330 PRINT
340 PRINT"    DOVE CODICE CARATTERE (0-255)"
350 PRINT"         POSIZIONE X      (0-75)"
360 PRINT"         SET              (0-1)"
370 PRINT
380 PRINT" PREMI UN TASTO PER INIZIARE ....."
390 GETKEYA$
400 :END
1000 DATA32,188,190,226,172,225,191,251,187,255,161,236,162,254,252,96,169,208
1100 DATA133,4,173,251,0,41,1,240,4,169,216,133,4,169,0,162,3,6,250,42,202,208
1200 DATA250,24,101,4,133,4,165,250,133,3,169,0,133,250,169,5,133,2,160,0,234
1300 DATA234,234,234,32,224,11,133,5,230,3,234,234,234,234,234,234,234,32
1400 DATA224,11,133,6,230,3,198,2,240,28,162,4,169,0,6,6,42,6,6,42,6,5,42,6,5
1500 DATA42,164,250,153,16,12,230,250,202,208,232,240,196,76,132,11,0,0,0,0,0
1600 DATA0,0,0,0,0,0,160,0,166,249,240,8,169,29,32,210,255,202,208,250,169,4,133
1700 DATA6,185,16,12,170,189,0,11,133,5,41,64,240,5,169,18,32,210,255,165,5,41
1800 DATA191,32,210,255,169,146,32,210,255,200,198,6,208,221,169,13,32,210,255
1900 DATA192,16,208,196,234,104,133,2,104,133,3,104,133,4,104,133,5,104,133,6
2000 DATA88,96,133,250,134,249,132,251,76,239,11,0,255,0,142,223,11,169,3,162
2100 DATA14,32,116,255,174,223,11,96,104,120,165,2,72,165,3,72,165,4,72,165,5
2200 DATA72,165,6,72,76,16,11
2300 END
```

```
10 REM PRG PER C/128. CONSENTE DI CARICARE ED ATTIVARE
20 REM SET DI CARATTERI PROGRAMMATI. QUESTI DEVONO ESSER
30 REM PRESENTI SU DISCO ED IL LORO NOME DEVE INIZIARE
40 REM CON "SET" PER CONSENTIRE IL CARICAMENTO
50 REM AUTOMATICO DEI LORO NOMI
60 :
100 GOTO 200
110 OPEN 1,8,0,"$:SET*":DIM V$(144)
120 GET#1,A$:GET#1,A$:FAST
130 GET#1,L$:GET#1,L$
140 IF ST=64 THEN CLOSE1:SLOW:RETURN
150 GET#1,LB$:GET#1,HB$:LN=ASC(LB$)+256*ASC(HB$)
```

```
160 V$(Q)=V$(Q)+STR$(LN)
170 GET#1,A$:IFA$=""THEN Q=Q+1:GOTO130
180 IFA$=CHR$(34)THENA$=" "
190 V$(Q)=V$(Q)+A$:GOTO 170
200 SCNCLR
210 PRINT"ATTENDI..."
220 GOSUB 110
230 SCNCLR
240 PRINT
250 PRINT" DISCO :";LEFT$(RIGHT$(V$(0),23),16)
260 PRINTTAB(8)"----------------"
270 W=Q-2:IF W<0 THEN W=0
280 PRINT " FONT TROVATI:";:PRINT USING "###";W
290 PRINT
300 PRINT " SPOSTA CON I CURSORI ■ SPAZIO ■ ATTIVA "
320 PRINT
330 PRINT TAB(18)"-------------------"
340 PRINT TAB(18)"!                 !"
350 PRINT TAB(18)"-------------------"
360 CHAR 1,1,8,"FONT SELEZIONATO:"
370 PRINT:PRINT:PRINT
380 PRINT TAB(18)"-------------------"
390 PRINT TAB(18)"!                 !"
400 PRINT TAB(18)"-------------------"
410 CHAR 1,5,12,"FONT DI TIPO:":K=1
420 DO
430 GET KEY A$
440 IF ASC (A$)=145 THEN BEGIN
450 IF K=<1 THEN 480
460 K=K-1
470 CHAR 1,20,8,MID$(V$(K),7,16)
480 BEND
490 IF ASC(A$)=17 THEN BEGIN
500 IFK=Q-2 THEN 530
510 K=K+1
520 CHAR 1,20,8,MID$(V$(K),7,16)
530 BEND
540 IF VAL(LEFT$(V$(K),4))>9 THEN CHAR 1,20,12,"DOPPIO   ":D=2:ELSE CHAR 1,20,12
,"SINGOLO   ":D=1
550 LOOP UNTIL A$=" "
560 A$=MID$(V$(K),7,17)
570 Z=LEN(A$):DO
580 A$=LEFT$(A$,LEN(A$)-1)
590 Z=Z-1
600 LOOP UNTIL MID$(A$,Z,1)<>" "
610 CHAR 1,0,16,""
620 PRINT" ORA CARICO :■"A$:MD=RGR(0)
630 GRAPHIC1:GRAPHIC MD:BLOAD (A$),P8192:IF D=1 THEN BLOAD(A$),P10240
635 IF RGR(0)=0 THEN BEGIN
640 PRINT
650 PRINT" ■ F1 ■ PER RIATTIVARE IL SET"
660 PRINT
670 POKE 2604,24
680 KEY 1,"POKE2604,24"+CHR$(13)
690 BEND:ELSE BEGIN
700 FAST:COLOR6,1
710 FORI=0TO63:READX:POKEDEC("D81")+I,X:NEXT:GOTO790
730 DATA 169,0,160,32,133,218,132,219,162,18,169,32,32,204,205,232,169,0,32,204,
205,160,0,162,14,169,218,32,116,255,32,202,205
740 DATA 200,192,8,144,241,169,0,32,202,205,136,208,250,24,165,218,105,8,133,218
,144,224,230,219,165,219,201,48,144,216,96
750 POKEDEC("D15"),ADAND255
760 POKEDEC("D0F"),AD/256
770 POKEDEC("D1C"),BI
780 SYSDEC("D0C")
790 SYSDEC("D81"):SYS52332,,22,1
800 GRAPHIC CLR
810 BEND
820 REM: NEW
```

TANTO SOFTWARE PER IL TUO C/64

Presentiamo, anche se sommariamente, il contenuto di alcuni dischetti reperibili, da oggi, in tutte le edicole

Il numero speciale di "COMMODORE 64 CLUB" contiene una raccolta di utilities d'interesse generale; tra queste ricordiamo:

Menu Manager

è un programma per la creazione di menu su dischetto con cui personalizzare a piacimento le modalità di accesso ai files presenti nella directory, in modo da velocizzare notevolmente le operazioni di ricerca e caricamento.

Card Games

aggiunge al Basic 2.0 il nuovo, potente comando CARD che consente di gestire il mazzo di carte, di solito utilizzato nei giochi più noti (Scala Quaranta, Poker e cos via), in modo da realizzare complessi giochi di carte. Sul dischetto sono presenti due giochi-esempio: la Briscola ed il Black Jack, che potrete esaminare e modificare a piacimento.

Basic New

Naturale evoluzione del Super Extender, il Basic New risponde alle esigenze di coloro

che faticavano ad utilizzare la vecchia versione, per mancanza della necessaria esperienza. Tale utility serve per la personalizzazione del Basic 2.0 mediante l'aggiunta di diverse routine (tratte, ad esempio, dalle routine dell'enciclopedia in l.m. pubblicata regolarmente sulle nostre pagine).

Tutte le operazioni sono facilitate da un'interfaccia utente completamente user-friendly; la gestione a finestre multiple, infatti, consente una veloce e sicura gestione del nuovo interprete Basic. Il Basic New provvede anche alla generazione, mediante apposito linker, di un file eseguibile contenente il nuovo interprete, le routine aggiunte e qualche istruzione di gestione del sistema.

La presenza di alcuni programmi dimostrativi completa Basic New che dimostra di avere ptenzialit a dir poco sorprendenti. Sebbene a prima vista un programma simile possa sembrare riservato agli addetti ai lavori, possiamo dire con assoluta certezza che la presenza di una complessa interfaccia utente faciliti tutte le operazioni necessarie alla costruzione di un proprio interprete; cio alla portata di tutti.

Cruncher

è un programma di utilit non certo nuovo nel suo genere ma che presenta una sostanziale differenza dai suoi fratelli maggiori: la versione proposta utilizza un rivoluzionario algoritmo di compattazione che ottimizza lo spazio in memoria scegliendo fra quattro diversi modelli di riduzione.

Sector Editor

la potenza di questa utility consiste nella possibilit di accedere direttamente alla struttura logica dei dischetti, in modo da variare a piacimento il contenuto dei settori presenti sul disco.

Altri file di notevole interesse sono ovviamente presenti sul numero speciale "Utilities 2".

AMIGA VOICE EMULATOR

ovvero, come far parlare (e cantare) il C/64.

Il Commodore Amiga non l'unico computer in grado di sintetizzare la voce umana; anche il tuo C/64 possiede tale straordinaria caratteristica. Infatti possibile impartire istruzioni dedicate alla gestione dei fonemi (che sono le unità base di pronuncia) italiani semplicemente caricando un nuovo interprete Basic in grado di riconoscerle.

L'interprete di cui stiamo parlando aggiunge al Basic 2.0 i due nuovi comandi VOX e PLAY con i quali possibile impostare il timbro della voce, la velocità di lettura dei fonemi, che ha il compito di influire sulle "caratteristiche" della persona che pronuncia la frase impostata (aumentando la velocit il computer assume un tono decisamente femminile) e, ovviamente, la frase da pronunciare.

C'è comunque una caratteristica che differenzia questo prodotto dal comando Play implementato sull'Amiga: utilizzando in modo appropriato il comando Play possibile fare in modo che il C/64 canti!
Le possibilità offerte da Amiga voice emulator sono molteplici e trovano una completa applicazione in tutti quei programmi che richiedono un certo "dialogo" con l'utente.

Onde evitare spiacevoli equivoci, ricordiamo che "Amiga voice emulator" rappresenta la ristampa del famoso "La Voce 2", che, in precedenza, tanto successo ha ottenuto presso gli utenti del C/64-128. Il dischetto viene quindi consigliato soprattutto ai nuovi utenti del piccolo Commodore, oppure ai neo acquirenti del drive 1541 che, quando possedevano il solo registratore, non hanno potuto utilizzare "La Voce 2" proposta, in precedenza, su disco.

COMMODORE 64 CLUB #9

Puntuali, come sempre, all'appuntamento in edicola, presentiamo il nuovo numero della serie "Commodore 64 Club" che offre al lettore la possibilità di acquistare il vero Software made in Italy.
In questo numero troverete:

Rings'n'up

è una nuova avventura per il simpatico eroe di Mr. Hat, che si trova imprigionato in paurosi labirinti popolati da orrendi mostri, alla ricerca di tesori, fama e gloria.

Nell'episodio precedente avevamo lasciato il nostro eroe alla ricerca del magico cappello d'oro; una volta trovato tale oggetto il buon Mr. Hat dovrà uscire dal labirinto in cui si trova, raccogliendo un certo numero di chiavi ed anelli che servono per aprire le porte che trover lungo il cammino. E' appunto questa la fase che caratterizza il gioco Rings'n'up; condurre Mr. Hat in salvo.

Aurora

Alla guida di un potente mezzo da battaglia dovrete penetrare le difese nemiche dislocate nel bosco Aurora ed annientare il generatore che controlla l'emissione del mortale raggio distruttivo che eliminerà la razza umana dalla galassia.

Triacus

In questo gioco vi troverete, alla guida di un caccia spaziale, alla conquista dei pianeti criminali, responsabili della distruzione totale di alcuni sistemi stellari. Dovrete penetrare le difese del pianeta ed eliminare tutte le astronavi da ricognizione che troverete. A quel punto il pianeta esploderà e la vostra missione sarà completa.

Briscola e Black Jack

Sono i due classici giochi di carte che chiunque avrà avuto modo di osservare sui tavoli delle osterie di campagna; i programmi sono stati realizzati utilizzando la routine Card Games presente sul disco Utilities 2, descritto in queste stesse pagine.

Calendar

Questa simpatica utility serve per visualizzare, ed eventualmente stampare, un calendario relativo ad un mese specifico. I giorni della settimana sono calcolati utilizzando un algoritmo di calendario perpetuo che considera automaticamente anche tutti gli anni bisestili.

Melissa

Un gradito ritorno in edicola di una delle più famose avventure prodotte dalla Systems Editoriale. Sensibili, come sempre, alle sollecitazioni dei lettori che richiedono spesso la soluzione delle nostre avventure, abbiamo iniziato pubblicando, sul numero scorso, l'avventura Mission Twain; l'iniziativa continua su questo numero con la soluzione di Mission Twain e l'avventura Melissa.

TUTTI EROI CON "POOL OF RADIANCE"

Finalmente sul vostro Commodore 64/128 il primo e più diffuso gioco di ruolo

di **Wind**

Finora i 64-isti patiti dei giochi di ruolo hanno dovuto accontentarsi di pallide imitazioni di "Dungeon & Dragons"; ma, finalmente, è disponibile la versione originale!

Esaminato il software, possiamo affermare che valeva la pena aspettare così tanto tempo.

Parliamo, innanzitutto, degli autori del fantastico programma. "Pool of Radiance" è nato dalla collaborazione di esperti della TSR (la ditta produttrice dell'omonimo gioco da tavolo), di programmatori della SSI (specializzata in conversioni di giochi da tavolo e RPG) ed il talento produttivo della US Gold.

Il gioco è tratto dalla serie "Dragonlance" e tutto lascia supporre che questa non sarà l'ultima traduzione della SSI, considerata la sensazionale accoglienza riscossa in America dall'iniziativa che mira a portare il gioco da tavolo più diffuso nel mondo (più diffuso dello stesso Monopoli) interamente su computer.

In contemporanea al gioco di cui ci stiamo occupando, infatti, ne sono stati presentati altri due: "Heroes of the Lance" (del genere arcade) e "Dungeon Master Assistant" (una utility); il primo accontenta i patiti dell'azione, il secondo aiuta il giocatore nel difficile compito di "Dungeon Master".

Tralasciamo ora questi particolari e passiamo a parlare in termini più approfonditi del programma.

UN RUOLO CHE VI SI ADDICE

Nel gioco rappresentate un gruppo di avventurieri che hanno intravisto, nell'avviso affisso sulle mura della propria città, l'opportunità di arricchirsi e diventare famosi mostrando a tutti il proprio valore.

Giunti nei Reami Dimenticati, luogo ove l'avventura avrà luogo, siete accolti nel porto di Phlan da un inviato del Consiglio (formato dai mercanti che hanno deciso di portare a nuova vita la città invasa quasi totalmente dai mostri). Il personaggio vi conduce nei luoghi più noti per farvi ambientare; dopo aver visto i dintorni, vi lascia davanti all'ingresso che porta alla parte vecchia di Phlan, un luogo che vi attrae e vi spaventa e che dovete ripulire dalle creature che vi si annidano in agguato, pronte ad uccidervi se non starete attenti.

Se pensate che entrare da soli sia troppo pericoloso potete andare in giro per la città e cercare altre persone coraggiose con cui dividere pericoli (e tesori).

Se pensate, invece, di essere in grado di iniziare senza problemi, comprerete, con i risparmi di tutta la vostra vita, le armi e le attrezzature che pensate siano necessarie. Se siete, però, dei principianti, e non avete mai giocato a D&D non preoccupatevi: nella confezione originale trovate indicazioni utili per compiere le prime mosse oltre ad alcuni suggerimenti sufficienti per superare le prime fasi (che sono le più difficoltose) in modo da proseguire speditamente alla ricerca della mitica "Pozza Irradiante".

Alla confezione è accluso perfino un libretto (molto ben fatto) che illustra la storia di Phlan e le vicissitudini che hanno interessato la cittadina dal momento in cui è stata fondata fino al vostro arrivo.

Nel volumetto troverete anche il diario dell'avventuriero in cui sono riportate varie notizie, alcune vere ed altre false, che potreste percepire durante il vostro peregrinare: alcune di queste voci sono vere, mentre altre non lo sono e scoprirete, durante l'avventura, quali fra queste sono degne di fede.

Il manuale di gioco è molto utile per chi non ha mai giocato a D&D dal momento

che non spiega nei dettagli i meccanismi del gioco, ma solo quella sua parte che interessa, tralasciando tutte le regole complesse (che rendono il gioco molto verosimile) e delle quali si occupa il computer che interpreta quello che nel gioco da tavolo è il DM.

A questo punto è d'obbligo sottolineare che le regole non sono esattamente quelle di D&D, ma quelle di AD&D (Advanced Dungeons & Dragons) le quali comprendono le normali regole con poche modifiche per rendere il gioco più realistico.

Ah... quasi dimenticavamo: sono presenti ben quattro floppy disk, per un totale di otto facciate, su una delle quali è presente un esempio di avventura. I dischetti non sono protetti, quindi potete effettuare copie di sicurezza; dovete poi procurarvi un quinto dischetto su cui memorizzare i progressi che farete procedendo nella ricerca.

Ma non illudetevi: esiste una protezione per evitare che il programma venga duplicato e rivenduto. Si tratta di una ruota decodificatrice senza la quale non è possibile arrivare alla soluzione.

PRIMA DI INIZIARE

Passiamo ora ad esaminare la meccanica del programma. Potete iniziare con dei personaggi già registrati (sul primo dischetto) oppure creare "vostri" personaggi. In quest'ultimo caso vengono poste alcune domande, la prima delle quali riguarda il "tipo" del personaggio da interpretare: si può scegliere fra piccoli e burberi nani, potenti elfi, buffi gnomi, misteriosi mezzi-elfi, simpatici halfling e, ovviamente, personaggi semplicemente umani.

Dovete poi impostare sesso e classe del personaggio stesso; si può scegliere tra quattro classi: chierico, guerriero, ladro, mago; potete sceglierne una o più, a seconda della razza. Ad esempio un umano può scegliere una sola classe ed essere, ad esempio, un mago. Un mezzo-elfo può invece selezionare fino a tre classi ed essere quindi un chierico / guerriero / mago.

Ogni avventuriero è inoltre identificato da sei caratteristiche: forza, intelligenza, saggezza, destrezza, costituzione e carisma. Ultimo, ma non meno importante, è l'allineamento, il quale definisce il vostro comportamento nei confronti degli altri esseri che incontrate.

La procedura rispetta le regole del gioco originale, ma a questo punto è possibile "personalizzare" l'eroe così costruito attribuendogli un volto ed un'armatura appropriata alla classe selezionata.

In seguito modificate l'icona che vi rappresenta durante la fase di combattimento, cambiandone la fisionomia, le armi, l'armatura, la posizione da assumere durante il combattimento, i colori ed altro ancora.

E' intuitivo che conviene creare più di un personaggio in quanto non è facile sopravvivere da soli, e i personaggi che si uniranno a voi non sempre faranno quello che voi volete.

Raggruppato un certo numero di avventurrieri potete cominciare il gioco.

NEL GIOCO

Quando vi muovete in città avete una visione tridimensionale dell'ambiente che vi circonda. Potete comunque easaminare la mappa delle vicinanze, se siete in una zona nota.

Lo schermo è diviso in tre parti: in basso è presente il menu di movimento, a sinistra la vista 3D / mappa e a destra lo stato del gruppo.

Il video cambia quando combattete contro qualcuno; le scelte del menu sottostante vi permettono di muovervi, attaccare, lanciare incantesimi, aiutare un compagno in difficoltà e così via; la vista in 3D effettua una zoomata sul personaggio che deve muovere mentre a destra vengono visualizzati i suoi dati principali.

Nelle zone esterne alla città viene mostrata direttamente la mappa delle zone adiacenti, molto più dettagliata di quelle precedenti.

Noi della redazione (dopo nottate passate evitando "trolls" e raccogliendo tesori) siamo riusciti a rintracciare l'antico drago d'argento, una delle possibili cause della rovina di Phlan, e contiamo di avere la soluzione prima che voi leggiate queste righe. Ogni vostro suggerimento sarà quindi gradito, così come saranno gradite eventuali vostre domande in merito.

In conclusione il prodotto rappresenta sicuramente una pietra miliare per il C/64 tra i "role playing" visto che unisce una semplicità d'uso ed un'accuratezza di gioco veramente notevole ad un prezzo decisamente adeguato.

Ne consigliamo quindi l'acquisto a tutti coloro che vogliono usare il computer per giocare in modo intelligente e non solo per sparare ai marzianetti.

"Pool of Radiance" è distribuito da: LEADER Distribuzione s.r.l. via Mazzini, 15 - 21040 Casciago (VA) - al prezzo di sole L. 59.000

QUANTO COSTA IL TUO COMMODORE

Amiga 2000 - L. 2.715.000

Microprocessore Motorola MC68000 - Clock 7,16MHz - Kickstart ROM - Memoria RAM 1 MByte - 3 Chip custom per DMA, Video, Audio, I/O - 5 Slot di Espansione Amiga Bus 100 pin Autoconfig™ - 1 Slot di Espansione 86 pin per Schede Coprocessore - 2 Slot di Espansione compatibili AT/XT - 2 Slot di Espansione compatibili XT - 2 Slot di Espansione Video - 1 Floppy Disk Drive da 3 1/2", 880 KBytes - Porta seriale RS232C - Sistema Operativo single-user, multitasking AmigaDOS - Compatibilità MS-DOS XT/AT disponibile con schede interne Janus (A2088 - A2286) - Monitor escluso

Amiga 500 - L. 995.000

Microprocessore Motorola MC68000 - Clock 7,16 MHz - Kickstart ROM - Memoria RAM 512 KBytes - 3 Chip custom per DMA, Video, Audio I/O - 1 Floppy Disk Driver da 3 1/2", 880 KBytes - Porta seriale RS232C - Porta parallela Centronics

Videomaster 2995 - L. 1.200.000

Desk Top Video - Sistema per elaborazini video semiprofessionale composto da genlock, digitalizzatore e alloggiamento per 3 drive A2010 - Ingressi videocomposito (2), RGB - Uscite Videocomposito, RF, RGB + sync -

Floppy Disk Driver A 1010 - L. 335.000

Floppy Disk Driver - Drive esterno da 3 1/2" - Capacità 880 KBytes - Collegabile a tutti i modelli della linea Amiga, alla scheda A2088 e al PC1

Floppy Disk Drive A 2010 - L. 280.000

Floppy Disk Drive - Drive interno aggiuntivo da 3 1/2" - Capacità 880 KBytes - Collegabile ad Amiga 2000

Hard Disk A 590 - L. 1.750.000

Hard Disk + Controller + RAM - Scheda Controller - Hard Disk da 3 1/2" 20 MBytes - 2 MBytes "fast" RAM - Collegabile all'Amiga 500

Scheda Janus A 2088 + A 2020 - L. 1.050.000

Scheda Janus XT + Floppy Disk Drive da 5 1/4", 360 KBytes - Scheda Bridgeboard per compatibilità MS-DOS (XT) in Amiga 2000 - Microprocessore Intel 8088 - Coprocessore matematico opzionale Intel 8087

A2286+A2020 - L. 1.985.000

Scheda Janus AT + Floppy Disk Drive da 5 1/4", 1,2 MBytes - Scheda Bridgeboard per compatibilità MS-DOS (AT) in Amiga 2000 - Microprocessore Intel 80287 - Clock 8 MHz - RAM: 1 MBytes on-board - Floppy Disk Controller on board - Floppy Disk Driver disegnato per l'installazione all'interno dell'Amiga 2000 -

Scheda A2620 - L. 2.700.000

Scheda Processore Alternativo 32 bit - Scheda per 68020 e Unix - Microprocessore Motorola MC68020 - Coprocessore matematico Motorola MC68881 (opzionale MC68882)

Scheda A Unix - L. 3.250.000

Sistema Operativo AT&T Unix System V Release 3 - Per Amiga 2000 con scheda A2620 e Hard Disk 100 MBytes

Hard Disk A2092+PC5060 - L. 1.020.000

Hard Disk e controller - Hard Disk 3 1/2" ST506 - Capacità formattata 20 MBytes

Hard Disk A2090+2092 - L. 1.240.000

Hard Disk e controller - Hard Disk 3 1/2" ST506 - Capacità formattata 20 MBytes

Hard Disk A2090+A2094 - L. 1.900.000

Stesse caratteristiche del kit A2092 ma con disco da 40 MBytes

Espansione di memoria A2058 - L. 1.149.000

Espansione di memoria - Scheda di espansione per Amiga 2000 - Fornita con 2 MBytes "fast" RAM, espandibile a 4 o 8 MBytes

Scheda Video A2060 - L. 165.000

Modulatore video - Scheda modulatore video interna per Amiga 2000 - Uscite colore e monocromatica - Si inserisce nello slot video dell'Amiga 2000

Genlock Card A2301 - L. 420.000

Genlock - Scheda Genlock semiprofessionale per Amiga 2000 - Permette di miscelare immagini provenienti da una sorgente esterna con immagini provenienti dal computer

Professional Video Adapter Card A2351 - L. 1.500.000

Professional Video Adapter - Scheda Video Professionale per Amiga 2000 (B) - Genlock qualità Broadcast - Frame Grabber - Digitalizzatore - Include software di controllo per la gestione interattiva
(Disponibile da maggio '89)

A501 - L. 300.000

Espansione di memoria - Cartuccia di espansione di memoria da 512 KBytes per A500

A520 - L. 45.000

Modulatore RF - Modulatore esterno A500 - Permette di connettere qualsiasi televisore B/N o colori ad Amiga 500

A Scart - L. 28.000

Cavo di collegamento A500/A2000 con connettore per televisione SCART

Monitor a colori 1084 - L. 615.000

Monitor a colori ad alta risoluzione - Tubo 14" Black Matrix antiriflesso - Pitch 0.39 mm - Compatibile con Amiga 500/2000, PC (tutta la gamma), C64 e C128

Monitor a colori 2080 - L. 770.000

Monitor a colori ad alta risoluzione e lunga persistenza - Tubo 14" Black Matrix antiriflesso - Pitch 0.39 mm - Frequenza di raster 50 Hz - Compatibile con Amiga 500/2000, PC (tutta la gamma), C64 e C128

Monitor Monocromatico A2024 - L. 1.235.000

Monitor monocromatico a fosfori "bianco-carta" - Turbo 14" antiriflesso - (Disponibile da marzo '89)

PC60/40 - L. 8.930.000

Microprocessore Intel 80386 - Coprocessore matematico opzionale Intel 80387 - Clock 8 o 16 MHz selezionabile via software e da tastiera - Monitor monocromatico 14" - Tastiera avanzata 102 tasti con 12 funzioni - Sistema Operativo MS-DOS 3 2.1 - Interprete GW-Basic

PC60/40C - L. 9.180.000

Stessa configurazione ma con monitor 14" a colori mod. 1084

PC 60/80 - L. 10.450.000

Microprocessore Intel 80386 - Coprocessore opzionale Intel 80387 - Clock 8 o 16 MHz selezionabile via software e da tastiera - Memoria RAM: 2.5 MBytes - 1 Floppy Disk Drive da 5 1/4", 1.2 MBytes - 1 Floppy Disk Drive opzionale da 3 1/2", 1.44 MBytes - 1 Hard Disk da 80 MBytes - 2 Porte parallele Centronics - Mouse video EGA (compatibile MDA - Hercules - CGA). Emulazioni disponibili via hardware e software - Monitor monocromatico 14" - Tastiera avanzata 102 tasti con 12 tasti funzione - Sistema Operativo MS-DOS 3.21 - Ambiente Operativo Microsoft Windows/386 - Interprete GW-Basic

PC60/80C - L. 10.700.000

Stessa configurazione ma con monitor 14" a colori mod. 1084

PC40/20 - L. 4.100.000

Microprocessore Intel 80286 - Coprocessore matematico opzionale Intel 80287 - Clock 6 o 10 MHz selezionabile via software, hardware o da tastiera - Memoria RAM: 1 MByte - 1 Floppy Disk Drive da 5 1/4", 1.2 MBytes - 1 Hard Disk da 20 MBytes - Porta seriale RS232 - Porta parallela Centronics - Scheda video AGA multistandard (MDA - Hercules - CGA) Emulazioni disponibili via hardware e software - Monitor monocromatico 14" - Tastiera avanzata 102 tasti con 12 tasti funzione - Sistema Operativo MS-DOS 3.21 - Interprete GW-Basic

PC40/20C - L. 4.350.000

Stessa configurazione ma con monitor 14" a colori mod. 1084

PC 40/40 - L. 5.285.000

Microprocessore Intel 80286 - Coprocessore matematico opzionale Intel 80287 - Clock 6 o 10 MHz selezionabile via software, hardware o da tastiera - Memoria RAM: 1 MByte - 1 Floppy Disk Drive da 5 1/4", 1.2 MBytes - 1 Hard Disk da 20 MBytes - Porta seriale RS232 - Porta parallela Centronics - Scheda video AGA multistandard (MDA - Hercules - CGA) Emulazioni disponibili via hardware e software - Monitor monocromatico 14" - Tastiera avanzata 102 tasti con 12 tasti funzione - Sistema Operativo MS-DOS 3.21 - Interprete GW-Basic

PC40/40C - L. 5.535.000

Stessa configurazione ma con monitor 14" a colori mod. 1084

1352 - L. 78.000

Mouse - Collegabile con Microsoft Bus Mouse - Collegabile direttamente a PC1, PC10/20 - III, PC40 - III

PC910 - L. 355.000

Floppy Disk Drive - Drive interno aggiuntivo da 3 1/2" per PC10/20-I-II-III - Capacità 360 o 720 KBytes selezionabile tramite "config. sys" - Corredo di telaio di supporto per l'installazione in un alloggiamento per un drive da 5 1/4" - Interfaccia identica ai modelli da 5 1/4"

PC1 - L. 995.000

Microprocessore Intel 8088 - 1 Floppy Disk Drive da 5 1/4" - Porta seriale RS232C - Porta parallela Centronics - - Monitor monocromatico 12" - Tastiera 84 tasti - Sistema Operativo MS-DOS 3.2 - Interprete GW-Basic

PCEXP1 - L. 640.000

PC Expansion Box - Box esterno di espansione per PC 1 - Alimentatore aggiuntivo incluso - Contiene 3 Slot di Espansione compatibili Ibm XT - Alloggiamento per Hard Disk da 5 1/4" - Si posiziona sotto il corpo del PC1 e viene collegato tramite degli appositi connettori

PC10-III - L. 1.965.000

Microprocessore Intel 8088 Clock 4.77 MHz 9.54 MHz (double) selezionabile via software e da tastiera - Memoria RAM: 640 KBytes - 2 Floppy Disk Drive da 5 1/4", 360 KBytes - Porta seriale RS232C - Porta parallela Centronics - Porta Mouse per Mouse Commodore 1352 (compatibile Microsoft Bus Mouse - Tastiera avanzata 102 con 12 tasti funzione Sistema Operativo MS-DOS 3.21 - Interprete GW-Basic

PC10-IIIC - L. 2.300.000

Stessa configurazione ma con monitor 14" a colori mod. 1084

PC20-III - L. 2.715.000

Microprocessore Intel 8088 - Clock 4.77 MHz 9.54 MHz (double) selezionabile via software e da tastiera - 1/4", 360 KBytes - 1 Hard Disk da 20 MBytes - Porta seriale RS232C - Porta parallela Centronics - Porta Mouse per Mouse Commodore 1352 (compatibile Microsoft Bus Mouse) - Tastiera avanzata 102 con 12 tasti funzione Sistema Operativo MS-DOS 3.21 - Interprete GW-Basic

PC20-IIIC - L. 3.050.000

Stessa configurazione ma con monitor 14" a colori mod. 1084

Nuovo C64 - L. 325.000

Nuovo Personal Computer CPU 64 KBytes RAM - Vastissima biblioteca software disponibile - Porta seriale Commodore - Porta registratore per cassette - Porta parallela programmabile -

C128D - L. 895.000

Personal Computer CPU 128 KBytes RAM espandibile a 512 KBytes - ROM 48 KBytes - Basic 7.0 - Tastiera separata - Funzionante in modo 128,64 o CP/M 3.0 - Include floppy disk drive da 340 KBytes

Floppy Disk Drive 1541 II - L. 365.000

Floppy Disk Drive - Floppy Disk Drive da 5 1/4" singola faccia - Capacità 170 KBytes - Alimentazione separata - Compatibile con C64, C128, C128D

Floppy Disk Dirve 1581 - L. 420.000

Floppy Disk Drive da 3 1/2" doppia faccia - Capacità 800 KBytes - Alimentazione separata - Compatibile con C64, C128, C128D

1530 - L. 55.000

Registratore a cassette per C64, C128, C128D

Accessori per C64 - 128D

1700 - Espansione di memoria - Cartuccia di espansione di memoria a 128 KBytes per C128 - **L. 170.000**

1750 - Espansione di memoria - Cartuccia di espansione di memoria 512 KBytes per C128 - **L. 245.000**

1764 - Espansione di memoria - Cartuccia di espansione di memoria a 256 KBytes per C64
Fornita di alimentatore surdimensionato - **L. 198.000**

16499 - Adattatore Telematico Omologato - Collegabile al C64
Permette il collegamento a Videotel, P.G.E. e banche dati - **L. 149.000**

1399 - Joystick - Joystick a microswitch con autofire - **L. 29.000**

1351 - Mouse - Mouse per C64, C128, C128D - **L. 72.000**

Monitor Monocromatico 1402 - L. 280.000

Monitor monocromatico a fosfori "bianco-carta" - Turbo 12" antiriflesso - Ingresso TTL - Compatibile con tutta la gamma PC

Monitor Monocromatico 1404 - L. 365.000

Monitor monocromatico a fosfori ambra - Turbo 14" antiriflesso a schermo piatto - Ingresso TTL - Compatibile con tutta la gamma PC - Base orientabile

Monitor Monocromatico 1450 - L. 470.000

Monitor monocromatico BI-SYNC a fosfori "bianco-carta" - Turbo 14" antiriflesso - Ingresso analogico e digitale - Doppia frequenza di sincronismo orizzontale per compatibilità con adattatori video MDA, Hercules, CGA, EGA e VGA

Monitor a colori 1802 - L. 445.000

Monitor a colori - Turbo 14" - Collegabile a C64, C128, C128D

Monitor monocromatico 1900 - L. 199.000

Monitor monocromatico a fosfori verdi - Turbo 12" antiriflesso - Ingresso videocomposito - Compatibile con tutta la gamma Commodore

Monitor a colori 1950 - L. 1.280.000

Monitor a colori BI-SYNC alta risoluzione - Turbo 14" antiriflesso - Ingresso analogico e digitale - Doppia frequenza di sincronismo orizzontale per compatibilità con adattatori video MDA, Hercules, CGA, EGA e VGA

Stampante MPS 1230 - L. 465.000

Stampante a matrice di punti - Testina a 9 aghi - 120 cps - Bidirezionale - 80 colonne - Near Letter Quality - Stampa grafica - Fogli singoli e modulo continuo - Trascinamento a trattore e/o frizione - Interfaccia seriale Commodore e parallela Centronics - Compatibile con tutti i prodotti Commodore

MPS 1230R - L. 19.000

Nastro per stampante

Stampante MPS 1500C - L. 550.000

Stampante a colori a matrice di punti - Testina a 9 aghi - 130 cps - Bidirezionale - 80 colonne - Supporta nastro a colori o nero - Near Letter Quality - Stampa grafica - Fogli singoli e modulo continuo - Trascinamento a trattore e/o frizione - Interfaccia parallela Centronics - Compatibile con la gamma Amiga e PC

MPS1500R - L. 37.000

Nastro a colori per stampante

Stampante MPS 1550C - L. 575.000

Stampante a colori a matrice di punti - Testina a 9 aghi - 130 cps - Bidirezionale - 80 colonne - Supporta nastro a colori o nero - Near Letter Quality - Stampa grafica - Fogli singoli e modulo continuo - Trascinamento a trattore e/o frizione - Interfaccia seriale Commodore e parallela Centronics - Compatibile con tutti i prodotti Commodore

I COMMODORE POINT

LOMBARDIA

Milano

- AL RISPARMIO - V.le Monza, 204
- BCS - Via Montegani, 11
- BRAHA ALBERTO
- Via Pier Capponi, 5
- E.D.S. - Corso Porta Ticinese, 4
- E.S.C. - Via Roggia Scagna, 7
- FAREF - Via A. Volta, 21
- FLOPPERIA - Viale Monte Nero, 31
- GBC - Via Cantoni, 7
- Via Petreila, 6
- GIGLIONI LAURA - Via D'Ovidio, 8
- GILGLIONI - Viale Luigi Sturzo, 45
- LOGITEK - Via Golgi, 60
- MARC[illegible] - Via F.lli Bronzetti, 37
- MEL[illegible]HIONI - Via P. Colletta, 37
- MESSAGGERIE MUSICALI
- Galleria del Corso, 2
- NEWEL - Via Mac Mahon, 75
- RIVOLA - Via Vitruvio, 43

Provincia di Milano

- IL CURSORE - Via Cavour, 35
- Novate Milanese
- REC ELGRA - Corso Milano, 118
- Bovisio Masciago
- F.LLI GALIMBERTI
- Via Nazionale dei Giovi, 28/36
- Barlassina
- GBC - Viale Matteotti, 66
- Cinisello Balsamo
- P. GIORGIO OSTELLARI
- Via Milano, 300 - Desio
- CASA DELLA MUSICA
- Via Indipendenza, 21
- Cologno Monzese
- PENATI - Via Verdi, 28/30
- Corbetta
- EMP - Viale Italia, 12 - Corsico
- CENTRO COMPUTER PANDOLFI
- Via Corridoni, 18 - Legnano
- COMPUTEAM - Via Vecellio, 41
- Lissone
- FUTURA - Via Solferino, 31 - Lodi
- M.B.M. - Corso Roma, 12 - Lodi
- L'AMICO DEL COMPUTER
- Viale Lombardia, 17
- Melegnano
- BIT 84 - Via Italia, 4 - Monza
- I.C.O. - Via dei Tigli, 14 - Opera
- NIWA HARD & SOFT
- Via Bruno Buozzi, 94
- Sesto S. Giovanni
- FERRARI LUIGI - Via Madre Cabrini
- Sant'Angelo Lodigiano
- IL COMPUTER SERVICE SHOP
- Via Padana Superiore, 197
- Vimodrone

Bergamo

- COMIF - Via Autolinee, 10
- CORDANI - Via dei Caniana, 8
- D.R.B. - Via Borgo Palazzo, 65
- NEW SYSTEMS - Via Paglia, 36

Provincia di Bergamo

- BERTULEZZI G. - Via Fantoni, 48
- Alzano Lombardo
- COMPUTER TEAM - Via Verdi, 1/B
- Carvico
- OTTICO ROVETTA
- Piazza Garibaldi, 6 - Lovere
- A.I.S. INTERNATIONAL
- Via San Carlo, 25
San Pellegrino Terme
- SISTHEMA - Via Roma, 45
- Sarnico
- COMPUTER POINT
- Via Lantieri, 52 - Sarnico
- AB INFORMATICA
- Statale Cremasca, 66
- Urgnano

Brescia

- COMPUTER CENTER - Via Cipro, 6
- INFORMATICA 2000
- Via Stazione, 16/B
- MASTER INFORMATICA
- Via F.lli Ugoni, 10/B
- VIGASIO MARIO - Port. Zanardelli, 3

Provincia di Brescia

- MISTER BIT - Via Mazzini, 70 -Breno
- CAVALLI PIETRO
- Via 10 Giornate, 14/B
- Castrezzato
- VIETTI GIUSEPPE - Via Milano, 1/B
- Chiari
- MEGABYTE - Piazza Maluezzi, 14
- Desenzano del Garda
- CO-RE - Via XXV Aprile, 136 - Flero
- BARESI RINO & C.
- Via XX Settembre, 7 - Ghedi
- INFO CAM - Via Provinciale, 38
- Gratacasolo

Como

- IL COMPUTER - Via Indipendenza, 90
- 2M ELETTRONICA - Via Sacco, 3

Provincia di Como

- ELTRONGROS - Via L. da Vinci, 54
- Barzanò
- EGÁ - Via Mazzini, 42
- Cassago Brianza
- DATA FOUND - Via A. Volta, 4 -Erba
- EGA - Via A. Moro, 17 - Galbiate
- RIGHI ELETTRONICA
- Via G. Leopardi, 26
- Olgiate Comasco
- CIMA ELETTRONICA
- Via Leonardo da Vinci, 7
- Lecco
- FUMAGALLI - Via Cairoli, 48 -Lecco

Cremona

- MONDO COMPUTER
- Via Giuseppina, 11/B
- PRISMA - Via Buoso sa Dovara, 8
- TELCO - Piazza Marconi, 2/A

Provincia di Cremona

- ELCOM - Via IV Novembre, 56/58
- Crema
- EUROELETTRONICA
- Via XX Settembre, 92/A - Crema

Mantova

- TUBALDO ELSO & C.
- Galleria Fermi, 7
- 32 BIT - Via Cesare Battisti, 14
- ELETTRONICA BASSO
- V.le Risorgimento, 69

Pavia

- POLIWARE - C.so C. Alberto, 76

Provincia di Pavia

- LOGICA INFORMATICA
- Via Monte grappa, 32 - Vigevano
- M. VISENTIN - C.so V. Emanuele, 76
- Vigevano

Sondrio

- CIPOLLA M. - Via Tremogge, 25

Provincia di Sondrio

- FOTONOVA - Via Valeriana, 1
- San Pietro di Berbenno

Varese

- DIMECO SISTEMI - Via Garibaldi
- IL CENTRO ELETTRONICO
- Via Morazzone, 2
- SUPERGAMES - Via Carrobbio, 13

Provincia di Varese

- BUSTO BIT - Via Gavinana, 17
- Busto Arsizio
- CRESPI G.&C. - V.le Lombardia, 59
- Castellanza
- COMPUTER SHOP
- Via A. da Brescia, 2 - Gallarate
- LIMA IMPORT-EXPORT
- (Grandi Magazzini Bossi)
- Via Clerici, 196 - Gerenzano
- J.A.C. - Via Matteotti, 38
- Sesto Calende

VALLE D'AOSTA

Aosta

- F.lli GATTI - Via Festaz, 75

PIEMONTE

Alessandria

- BIT MICRO - Via Mazzini, 104
- SERVIZI INFORMATICI
- Via Alessandro III, 47

Provincia di Alessandria

- S.G.E. ELETTRONICA
- Via Bandello, 19 - Tortona

Asti

- RECORD - Corso Alfieri, 166/3

Cuneo

- ROSSI COMPUTER - C.so Nizza, 42
- STUDIO Software - C.so Nizza, 49

Provincia di Cuneo

- PUNTO BIT - Corso Langhe, 26/C
- Alba
- SDI - Via Vittorio Emanuele, 250
- Bra
- ASCHIERI GIANFRANCO
- Corso Emanuele Filiberto, 6
- Fossano

Novara

- ELCOM - Corso Mazzini, 11
- PROGRAMMA 3 - V.le Buonarroti, 8
- PUNTO VIDEO
- Corso Risorgimento, 39/B

Provincia di Novara

- MIRCO POLACCO & C.
- Via Monte Zeda, 4 - Arona
- ALL COMPUTER
- Corso Garibaldi, 106
- Borgomanero
- MICROLOGIC - Via Giovanni XIII, 2
- Domodossola
- ELLIOT COMPUTER
- Via Don Minzoni, 32 - Intra

Torino

- ALBA ELETTRONICA
- Via C. Fossati, 5/P
- ALEX COMPUTER
- Corso Francia, 333/4
- COMPUTER HOME
- Via San Donato, 46/D
- COMPUTING NEWS
- Via Marco Polo, 40/E
- DE BUG - C.so V. Emanuele II, 22
- DESME UNIVERSAL
- Via San Secondo, 95
- F.D.S. - Via Borgaro, 86/D
- INFORMATICA ITALIA
- Corso Re Umberto, 128
- MT INFORMATICA
- Corso Giulio Cesare, 58
- MUSICS'S Shop - C.so Potenza, 177
- NEW BUSINESS COMPUTER
- Via Nizza, 45/F
- PLAY GAMES - Via C. Alberto, 39/A
- RADIO TV MIRAFIORI
- C.so Unione Sovietica, 381
- SMIT ELETTRONICA
- Via Bibiana, 83/B
- TELERITZ - Corso Traiano, 34

Provincia di Torino

- PAUL E CHICO VIDEOSOUND
- Via V. Emanuele, 52 - Chieri
- BIT INFORMATICA
- Via V. Emanuele, 154 - Ciriè
- HI-FI CLUB - Corso Francia, 92/C
- Collegno
- I.C.S. - Stradale Torino N. 73
- Ivrea
- BAS - Corso Roma, 47
- Moncalieri
- CERUTTI MAURO
- Corso Torino, 234 - Pinerolo
- EUREX Corso Indipendenza, 5
- Rivarolo C.se
- DIAM INFORMATICA
- C.so Francia, 146/Bis - Rivoli
- FULLINFORMATICA
- Via Vittorio Veneto, 25 - Rivoli
- GAMMA COMPUTER
- Via Cavour, 3 A/B
- Settimo Torinese

Vercelli

- ELETTROGAMMA - C.so Bormida, 27
- ELETTRONICA di BELLAMO A. & C.
- Strada Torino, 15

Provincia di Vercelli

- C.S.I. TEOREMA - Via Losana, 9
- Biella
- SIGEST - Via Bertodano, 8 - Biella
- REMONDINO Franco - Via Roma, 5
- Borgosesia
- FOTOSTUDIO TREVISAN
- Via XXV Aprile, 24/B
- Cossato
- STUDIO FOTOGRAFICO IMARISIO
- Piazza Martiri Libertà, 7 - Trino

LIGURIA

Genova

ABM COMPUTER
- Piazza De Ferrari, 24/R
- CENTRO ELETTRONICA
- Via Chiaravagna, 10/R
- Località Sestri Ponente
- COMMERCIALE SOTTORIPA
- Via Sottoripa, 115/117
- FOTOMONDIAL
- Via del Campo, 3-5-9-11-13/R
- LA NASCENTE - Via San Luca, 4/1
- RAPPR - EL - Via Boreoratti, 23/R

Imperia

- CASTELLINO - Via Belgrano, 44
- SASA COMPUTER
- Via Nazionale, 256

Provincia di Imperia

- CENTRO HI-FI VIDEO
- Via della Repubblica, 38 - Sanremo
- CASTELLINO - Via Genova, 48
- Ventimiglia

La Spezia

- CATTONI - Via Vitt. Veneto, 75
- I.L. ELETTRONICA
- Via Vitt. Veneto, 123

Provincia di La Spezia

- I.L. ELETTRONICA - Via Aurelia, 299
- Fornola di Vezzano

Savona

- CASTELLINO
- Corso Tardy e Benech, 101
- ATHENA
- Via Carissimo E. Crotti, 16/R

VENETO

Belluno

- UP TO DATE
- Via Vittorio Veneto, 43

Provincia di Belluno

- GUERRA COMPUTERS
- Viale Mazzini, 10/C - Feltre

Padova
- BIT SHOP - Via Cairoli, 11
- COMPUMANIA
- Riviera Tiso Camposanpiero, 37
- COMPUTER POINT - Via Roma, 63
- D.P.R. - V.le Lombardo, 4
- GIANFRANCO MARCATO
- Via Madonna della Salute, 51/53
- ZELLA ADELIO
- Piazza De Gasperi, 31/A

Provincia di Padova
- CAERT - Via Andorra, 11
- Zona Industriale Camin

Treviso
- BIT 2000 - Via Brandolini d'Adda, 14

Provincia di Treviso
- DE MARIN - Vai XX Settembre, 74
- Conegliano
- SIDESTREET - Via S. d'Acquisto, 8
- Montebelluna
- FALCON - Via Terraggio, 116
- Preganziol

Venezia
- TELERADIO FUGA
- San Marco, 3457

Provincia di Venezia
- GUERRA EGIDIO & C.
- Via Bissuola, 20/A - Mestre
- TREKILOWATT
- Via Torre Belfredo, 47 - Mestre
- REBEL - Via F. Crispi, 10
- San Donà di Piave
- GUERRA COMPUTERS
- Via Vizzotto, 29
- San Donà di Piave
- TELFERT - Via Chiesa, 1509
- Sottomarina
- RADIOCESTARO - Via Roma, 89
- Spinea

Verona
- CASA DELLA RADIO - Via Cairoli, 10
- TELESAT - Via Vasco de Gama, 8

Provincia di Verona
- FERRARIN - Via dei Massari, 10
- Legnago

Vicenza
- ELETTRONICA BISELLO
- Viale Trieste, 427/429
- SCLACHI Market - Via Cà Balbi, 139

Provincia di Vicenza
- GUERRA COMPUTERS
- Via Dell'Industria - Alte Ceccato
- SCIAVOTTO - Via Zanella, 21
- Cavazzale
- COMPUTER B. COSTO
- Via del Costo, 34 - Thiene
- ELETTROCASA - Via Roma, 67
- Trissino

FRIULI VENEZIA GIULIA

Gorizia
- E.C.O. ELETTRONICA
- Via F.lli Cossar, 23

Pordenone
- RIGO - Viale Cossetti, 5

Provincia di Pordenone
- MDT - Piazza Repubblica, 5
- Villanova di Prata
- BRUNO DA PIEVE
- Via Colombera, 17 - Porcia

Trieste
- AVANZO GIACOMO
- Piazza Cavana, 7
- COMPUTER SHOP - Via P. Reti, 6
- COMPUTIGI - Via XX Settembre, 51
- CTI - Via Pascoli, 4

Udine
- MOFERT 2 - Via Leopardi, 21
- R.T. SYSTEM - Via L. da Vinci, 99

Provincia di Udine
- IDRENO MATTIUSSI & C.
- Via Liciniana, 50 - Tavagnacco

Rovigo
- CLINICA DEL RASOIO
E DEL COMPUTER
- Via Fiume, 31/33

TRENTINO ALTO ADIGE

Bolzano
- C.M.B. ITALIA - Via Roma, 82
- MATTEUCCI PRESTIGE
- Via Museo, 54

Provincia di Bolzano
- ELECTRO TAPPEINER
- P.za Principale, 90 - Silandro
- RADIO MAIR - Via Centrale, 70
- Brunico
- ELECTRO RADIO HENDRICH
- Via delle Corse, 106 - Merano

Trento
- CRONST - Via Galilei, 25

EMILIA ROMAGNA

Piacenza
- COMPUTER Line - Via G. Carducci, 4
- DELTA COMPUTER
- Via Martiri della Resistenza, 15/G
- SOVER - Via IV Novembre, 60

TOSCANA

Arezzo
- DELTA SYSTEM - Via Piave, 13

Firenze
- ATEMA
- Via Bendetto Marcello, 1a/1b
- COOPERATIVA L.D.T. - Via Icaro, 9
- ELETTRONICA CENTOSTELLE
- Via Centostelle, 5/a
- HELP COMPUTER
- Via degli Artisti, 15/A
- PUNTO SOFT - Via Vagnetti, 17
- TELEINFORMATICA TOSCANA
- Via Bronzino, 36

Provincia di Firenze
- WAR GAMES
- Via Raffaello Sanzio, 126/A
- Empoli
- NEW E.V.M. COMPUTER
- Via degli Innocenti, 2
- Figline Valdarno
- COSCI F.LLI - Via Roma, 26 - Prato
- CENTRO INFORMATICA
- Via F.lli Cervi, 21/29 - Pontassieve

Grosseto
- COMPUTER SERVICE
- Piazza Ponchielli, 2

Livorno
- ETA BETA - Via San Francesco, 30
- FUTURA 2 - Via Cambini, 19

Provincia di Livorno
- ELETTRONICA ALESSI PAOLO
- Via Cimarosa, 1 - Piombino

Provincia di Lucca
- IL COMPUTER - V.le Colombo, 216
- Lido di Camaiore
- SANTI VITTORIO - Via Roma, 23
- San Romano Garfagnana

Massa
- EURO COMPUTER
- Piazza Bertagnini, 4
- FIRMWARE - Via Aurelia Ovest, 27

Provincia di Massa
- RADIO LUCONI - Via Roma, 24/B
- Carrara

Pisa
- C.H.S. - Via Carlo Cattaneo, 90/92
- ELECTRONIC SERVICE
- Via della Vecchia Tranvia, 10
- IT - LAB - Via Marche 8A/8B

Pistoia
- ELECTRONIC SHOP
- Via della Madonna, 49
- OFFICE DATA SERVICE
- Galleria Nazionale, 22

Provincia di Pistoia
- ZANNI & C. - Corso Roma, 45
- Montecatini Terme

Siena
- VIDEO MOVIE - Via Garibaldi, 17

Provincia di Siena
- ELECTRONIC Shop - Via A. Casini, 51
- Chianciano Terme
- ELETTRONICA
- Via di Gracciano nel Corso, 111
- Montepulciano

UMBRIA

Provincia di Perugia
- COMPUTER STUDIOS
- Via IV Novembre, 18/A
- Bastia Umbra

CAMPANIA

Provincia di Avellino
- FLIP FLOP - Via Appia, 68 - Atripalda

Benevento
- E.CO: INFORMATICA
- Via Pepicelli, 21/25

Caserta
- O.P.C. - Via G.M. Bosco, 24

Provincia di Caserta
- M.P. COMPUTER - Via Napoli, 30
- Maddaloni

Napoli
- BABY TOYS
- Via Cisterna dell'Olio, 5/Bis
- CASA MUSICALE RUGGIERO
- Piazza Garibaldi, 74
- CENTRO ELETTRONICO CAMPANO
- Via Epomeo, 121
- C.I.AN - Galleria Vanvitelli, 32
- DARVIN - Calata San Marco, 26
- ELETTRONICA RO.DA.LO.
- Via Epomeo, 216/B
- GIANCAR 2 - Piazza Garibaldi, 37
- GRUPPO BUSH
- Galleria Umberto I, 55
- ODORINO - Largo Lala, 22/A-B
- R 2 - Via F. Cilea, 285
- SPY - Via Fontana, 135
- TOP - Via S. Anna dei Lombardi, 12
- VIDEOFOTOMARKET
- Via S. Brigida, 19

Provincia di Napoli
- SPADARO - Via Romani, 93
- S. Anastasia
- TUFANO - S.S. Sannitica, 87 Km 7
- Casoria
- ELETTRONICA 2000
- Corso Durante, 40
- Frattamaggiore
- GATEWAY - Via Napoli, 68
- Mugnano
- NUOVA INFORMATICA SHOP
- Via Libertà, 185/191 - Portici
- BASIC COMPUTER
- C.so Garibaldi, 34
- Pozzuoli
- FALCO ELETTRONICA
- Via Sarno, 100 - Striano
- TECNOTRE - Via P. Fusco, 1/F
- Torre Annunziata

Salerno
- COMPUTER MARKET
- C.so Vitt. Emanuele, 23

Provincia di Salerno
- KING COMPUTER - Via Olevano, 56
- Battipaglia
- DIMER POINT - Via C. Rosselli, 20
- Eboli

PUGLIA

Bari
- ARTEL - Via G. d'Orso, 9
- COMPUTER'S ARTS
- Viale Meucci, 12/B

Provincia di Bari
- F. FAGGELLA - Corso Garibaldi, 15
- Barletta
- G. FAGGELLA
- Via P. d'Aragona, 62/A - Barletta
- G. LONUZZO - Via Nizza, 21
- Castellana

Brindisi
- MARANGI & MICCOLI
- Via Prov. San Vito, 165

Provincia di Foggia
- IL DISCOBOLO - Via T. Solis, 15
- San Severo

Lecce
- BIT
- Via 95° Regg.to Fanteria, 87/89

Provincia di Lecce
- CEDOK INFORMATICA - Via Roma, 31
- Tricase

Matera
- GUADIANO ELECTRONICS
- Via Roma, 1

Taranto
- ELETTROJOLLY - Via de Cesare, 13
- TEA - Via Regina Elena, 101

CALABRIA

Catanzaro
- C. & G. COMPUTER - Via F. Acri, 28
- PAONE SAVERIO - Via F. Acri, 93/99

Provincia di Catanzaro
- COMPUTER HOUSE - Via Bologna
- Crotone
- OTTICA FOTO NELLO RUELLO
- C.so Vittorio Emanuele, 177
- Vibo Valentia

Cosenza
- SIRANGELO COMPUTER
- Via N. Parisio, 25

Provincia di Cosenza
- ELIGIO ANNICHIARICO & C.
- Via Roma, 21 - Castrovillari
- ALFA COMPUTER
- Via Nazionale, 341/A
- Corigliano Scalo
- ING. FUSTO SALVATORE
- Corso Nicotera, 99
- Lamezia Terme

Reggio Calabria
- CONTROL SYSTEM
- Via S. Francesco da Paola, 49/DE
- SYSTEM HOUSE
- Via Fiume ang. Palestino, 1

Provincia di Reggio Calabria
- COMPUTER SHOP
- Via Matteotti, 50/52 - Locri

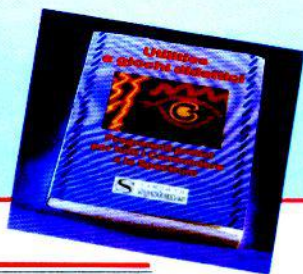
PRODOTTI SYSTEMS EDITORIALE

Software su cassetta

La voce III	L.12000
Raffaello	L.10000
Oroscopo	L.12000
Computer-Music	L.12000
Gestione familiare	L.12000
Banca dati	L.12000
Dichiarazione dei redditi (740/S)	L.16000
Matematica finanziaria	L.20000
Analisi di bilancio	L.20000
Arredare (richiede linguaggio Simon's Basic)	L.10000

Software su disco

Ms-Dos & Gw-Basic	L.25000
La voce III	L.12000
Oroscopo	L.12000
Gestione familiare	L.12000
Banca dati	L.12000
Dichiarazione dei redditi (740/S)	L.24000
Matematica finanziaria	L.20000
Analisi di bilancio	L.20000
Arredare (richiede linguaggio Simon's Basic)	L.20000
Graphic Expander C/128 in modo 80 colonne	L.27000

Offerta "Commodore speciale L.M." + dischetto — L.16000

Libri

64 programmi per il Commodore 64	L. 4800
I miei amici C/16 e Plus/4	L. 7000
Strategie vincenti per Commodore 64	L. 5800
62 programmi per Vic 20, C/16 e Plus/4	L. 6500
Tutti i segreti dello Spectrum	L. 7000
Impara giocando il Basic dello Spectrum	L. 7000
Micro Pascal per Commodore 64/128	L. 7000
Dal registratore al drive del C/64	L. 7000
Ada	L. 5000
Il linguaggio Pascal	L. 5000

Directory

Ciascun dischetto	L.12000

Arretrati

Ciascun numero arretrato di Commodore Computer Club 67	L. 5000
Ciascun numero arretrato di Personal Computer	L. 5000
Ciascun numero arretrato di VR Videoregistrare	L. 5000

Per un ottimale utilizzo del software "Linguaggio Macchina e Routine grafiche per C/64" è opportuna la lettura del fascicolo "Commodore Speciale" appositamente dedicato.

Coloro che desiderano procurarsi i prodotti della Systems Editoriale devono inviare, oltre alla cifra risultante dalla somma dei singoli prodotti, la cifra di L.3000 per spese di imballo e spedizione, oppure L.6000 se si preferisce la spedizione per mezzo raccomandata.

Sconti e agevolazioni

Le spese di imballo e spedizione sono a carico della Systems se ciascun ordine è pari ad almeno L.50000 (di listino).

Gli abbonati hanno diritto allo sconto del 10% e alla spedizione gratuita se la somma totale raggiunge la cifra di L.50000 (di listino).

Abbonamenti

Commodore Computer Club (11 fascicoli)	L. 50.000
Personal Computer (11 fascicoli)	L. 45.000
Commodore Computer Club + Personal Computer (11 + 11 fascicoli)	L. 95.000
VR Videoregistrare (12 numeri)	L. 50.000

N.B.: la cifra per gli abbonamenti non può essere conteggiata per ottenere gli sconti e le agevolazioni di cui sopra.

Compilate un normale modulo di C/C postale indirizzando a:

C/C postale N. 37952207
Systems Editoriale
Viale Famagosta, 75
20142 Milano

Non dimenticate di indicare chiaramente, sul retro del modulo (nello spazio indicato con "Causale del versamento") non solo il vostro nominativo completo di recapito telefonico, ma anche il materiale desiderato.

In ogni caso sarebbe opportuno inviare la presente scheda, debitamente compilata, allegando la fotocopia della ricevuta del versamento effettuato.

Chi volesse ricevere più celermente la confezione deve inviare la somma richiesta mediante assegno circolare oppure normale assegno bancario (non trasferibile o barrato due volte) intestato a: Systems Editoriale - Milano.

Commodore 64 Club

FANTASTICO

POINTER: TRASFORMA IL TUO 64 IN UN AMIGA E UTILIZZA IL JOYSTICK COME IL MOUSE

PUNKILLER: SFIDA AI GLADIATORI DEL 2000

é in edicola

IN OMAGGIO DA QUESTO NUMERO TUTTI I PROGRAMMI DI COMMODORE COMPUTER CLUB